Technische Universität Braunschweig
Carl-Friedrich-Gauß-Fakultät
Institut für Betriebssysteme und Rechnerverbund

Disruption Tolerant Vehicular Communication in Public Transportation Systems

Von der Carl-Friedrich-Gauß-Fakultät
Technische Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung des Grades

Doktor-Ingenieur (Dr.-Ing.)

genehmigte

Dissertation

von Michael Doering
geboren am 13.12.1978
in Braunschweig

Eingereicht am: 24.10.2012
Mündliche Prüfung am: 17.12.2012
Referent: Prof. Dr.-Ing. L. Wolf
Korreferent: Prof. Dr.-Ing. J. Ott

(2012)

Kurzfassung

Kommunikationssysteme leisten einen wichtigen Beitrag zum effizienten Betrieb des öffentlichen Personennahverkehrs. Seit einigen Jahren wird dabei der Sprechfunk zunehmend durch asynchronen M2M-Datenfunk ergänzt und in vielen Anwendungsgebieten sogar vollständig ersetzt. Die Kombination aus unterbrechungstoleranten Netzwerken und lizenzfreien Drahtlostechnologien birgt ein erhebliches Potential zur Reduzierung von Infrastrukturinvestitionen und Betriebskosten, da für diese Anwendungen eine dauerhafte Ende-zu-Ende Verbindung nicht mehr erforderlich ist. In dieser Arbeit wird die Machbarkeit eines solchen Systems untersucht und belegt. Zunächst werden dazu Anwendungsfälle vorgestellt und deren Anforderungen an das Kommunikationssystem analysiert. Dann werden die Kanalcharakteristika mehrerer WLAN-Technologien im realen ÖPNV-Umfeld experimentell ermittelt und bewertet. Auf Grundlage der erfolgversprechenden Ergebnisse werden im nächsten Schritt die besonderen Mobilitätseigenschaften von ÖPNV-Netzen untersucht. Zu diesen Zweck analysieren wir existierende und eigene, neu aufgezeichnete Bewegungsdaten von ÖPNV-Fahrzeugen. Unsere Daten enthalten dabei zusätzliche Metadaten der Verkehrsbetriebe, die zuvor nicht verfügbar waren, so dass wir unerwartete Effekte beschreiben und erstmals quantifizieren können. Anschließend werden die Bewegungsdaten mit den zuvor experimentell erfassten Kanaleigenschaften kombiniert, um so die Kommunikationskontakte zwischen den Fahrzeugen genauer zu betrachten. Wir stellen die statistische Verteilung der situationsabhängigen Kontaktereignisse vor, sowie den Einfluss der Funkreichweite auf die Kontaktkapazität. Dann werden die Ergebnisse aller vorhergehenden Schritte verwendet, um ein neues, optimiertes Routingverfahren für ÖPNV-Netze vorzuschlagen. In der simulationsbasierten Evaluation belegen wir, dass dieser Router die Leistung bisher bekannter Verfahren übertrifft.

Auf Grundlage der drei Hauptteile dieser Arbeit - Untersuchung der Kommunikationseigenschaften und Mobilitätsanalyse, Charakterisierung von Kommunikationskontakten, und Leistungsbewertung von Routingverfahren - stellen wir abschließend fest, dass das vorgeschlagene Kommunikationssystem die Anforderungen der Anwendungsfälle erfüllt. Weiterhin ist eine Umsetzung im großen Maßstab mit den bereits heute verfügbaren lizenzfreien Drahtlostechnologien möglich.

Abstract

Communication systems play an important role in the efficient operation of public transport networks. Recently, traditional voice-centric real-time communication is complemented and often replaced by data-centric asynchronous machine-to-machine communication. Disruption tolerant networking in combination with license-exempt high bandwidth technologies have the potential to reduce infrastructure investments and operating costs for such applications, because a continuous end-to-end connectivity is no longer required.

In this thesis the feasibility of such a system is investigated and confirmed. First, realistic use-cases are introduced and the requirements to the communication system are analyzed. Then the channel characteristics of several WLAN-based technologies are experimentally evaluated in real public transport scenarios. Since the results are promising, the next step is gaining a deeper understanding of the special mobility properties in public transport networks. Therefore, we analyze existing traces as well as our own newly acquired trace. Our trace features additional operator meta-data that is not available for existing traces, and we report on unexpected properties that have not been quantified before. Then the trace is combined with the experimentally obtained channel parameters in order to analyze the characteristics of inter-vehicle contacts. We present the statistical distribution of situation-specific contact events and the impact of radio range on contact capacity. Then results of all steps above are used to propose a routing scheme that is optimized for public transport networks. In the final simulation-based evaluation we show that this router outperforms previously proposed algorithms.

Based on the main three parts of this work - investigation of communication characteristics and mobility analysis, contact analysis, and routing - we conclude that the proposed communication system fulfills the requirements of current use-cases. Moreover, a large-scale implementation is feasible with license-exempt wireless technologies that are already available today.

Acknowledgments

During my work towards this thesis I experienced many challenges, some successes, a few dead ends, and a lot of inspiring discussions. I would like to thank my adviser Prof. Dr.-Ing. Lars Wolf for his constructive feedback and for creating an environment that was both productive and stimulating. Moreover, I thank Prof. Dr.-Ing. J. Ott for his agreement to be the assessor of this thesis.

My time at the Institut für Betriebssysteme und Rechnerverbund (IBR) was always more than “just work”. I met many great people, and some of those had a significant positive influence on my thesis. First, I would like to thank Dr. Marc Bechler and Dr. Sven Lahde for sparking my interest in research. Sven also taught me a lot about neat methodology and it was always a pleasure to work with him. Second, I would like to thank Dr. Jens Brandt for being such a great support for staying the course, and for his valuable and honest comments. Moreover, I thank Dr. Oliver Wellnitz for being the "experienced guy" and for many lively discussions. I also would like to thank Frank Steinberg, who always found a quick and unbureaucratic way to provide the insane computing, storage and networking capacity that was required for this thesis.

Further, I would like to thank Tobias Pögel and Bastian Pöttner for their contributions to my research. Special thanks also belongs to several students for supporting my work with their theses and projects, especially Till Lorentzen, Stephan Rottmann, Gerrit Lammert, Moritz Rosin, Hans-Peter Mohr and Nina Bergmann.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Communication systems have a significant impact on the efficiency and user-friendliness of public transport. Vehicle dispatching, traffic management, arrival prediction, dynamic passenger information and dynamic rerouting are just a few examples for processes that require communication. The increasing integration of information technology into the management and operations of public transport leads to a growing demand for wireless data transfer. Cellular data services provide sufficient range but relatively low data rates. Furthermore, the deployment and operation of a dedicated cellular infrastructure requires relatively high investments. For the operator of public transport, permanent cellular access costs might be inhibiting. An alternative is direct vehicle-to-vehicle communication based on license-exempt high bandwidth technologies such as IEEE 802.11a/b/g/n/p. These WLAN technologies are capable of significantly higher data rates but also have a lower range, so it is unlikely that each vehicle is constantly in the radio range of other vehicles. This problem could be solved by roadside infrastructure, which would again require high investments. A delay tolerant network (DTN) is a more economical approach because for many applications (e.g. asynchronous machine-to-machine communication, device monitoring, content updates) a slightly delayed delivery of data can be accepted.

Delay tolerant networking addresses the problem of connections that are often disrupted. End-to-end connections are not required in such a network. Instead, the protocol data units, referred to as bundles, may be stored until a connection to the next hop is available. This concept, also called store-carry-forward, is illustrated in figure 1.1. At first, a bundle for vehicle 3 is stored on vehicle 1, which moves and thereby carries the bundle along its trajectory (a). Then vehicle 1 gets within radio range of vehicle 2, and the bundle is forwarded via the radio link to vehicle 2 (b). Now, the vehicles move on and the connection breaks (c). Again, vehicle 2 carries the stored bundle along its way. Later, vehicle 2 encounters vehicle 3, and the bundle is finally delivered (d). In result, vehicle 1 was able to communicate with vehicle 3, although there never was a continuous end-to-end connection between vehicles 1 and 3.

According to this store-carry-forward concept, bundles are forwarded and stored again and again in a DTN, until they finally reach their destination. Choosing the 'right' sequence of next hops to minimize delay and maximize reliability is the routing algorithm's task. Several generic DTN routing algorithms have been proposed. These are suitable for a broad range of scenarios, but therefore cannot take advantage of special scenarios' properties, e.g. predetermined vehicle routes in public transport systems. However, DTNs are often deployed in very special scenarios and usually as "DTN-islands" with just a few edge-nodes operating as gateways to other DTN-islands or the Internet.

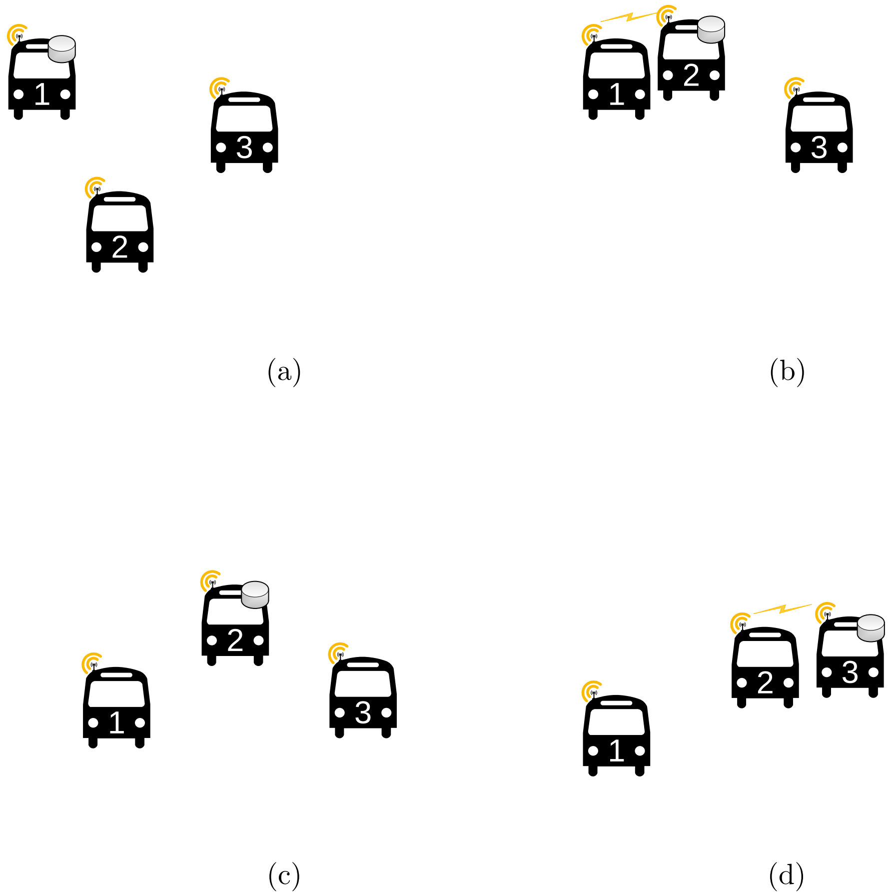

Figure 1.1: Store-Carry-Forward, the basic concept of disruption tolerant networking.

Moreover, because of DTN's store-carry-forward concept, it is possible to use multiple heterogeneous decentral routing algorithms simultaneously. Therefore, DTN-Islands can internally use more effective specialized routing, while edge-nodes use several different algorithms for interconnection.

This thesis holistically investigates the feasibility of a public transport communication system that builds upon license-exempt wireless technologies and disruption tolerant networking. Therefore, before finally proposing a routing algorithm optimized for public transport, several steps are required to gain a deeper understanding of the special properties of mobility and the communication properties in public transport networks.

For the development and implementation of a DTN-based communication system for public transport a suitable wireless technology has to be identified. License-exempt technologies are of special interest, because they can be used by public transport operators without renting radio spectrum. Therefore, the communication system would gain a significant cost advantage over existing infrastructure-based systems. However, the feasibility of license-exempt wireless technologies in urban vehicular applications has to be checked. The performance of these technologies - which were originally not designed for high mobility - has to be determined in dynamic DTN scenarios in which short contacts are typical. Provable realistic values for a feasibility study can only be gained by conducting field measurements in real-world scenarios.

A thorough investigation of disruption tolerance in public transport networks requires real-world mobility traces. These are indispensable for understanding the special mobility characteristics of public transport vehicles and the properties of DTN contacts. A suitable trace has to comprise vehicle position data as well as meta information like e.g. vehicle to line assignments, line definitions and timetables. This thesis also covers the acquisition and analysis of such a trace from one of the world's largest public transport network, because there were no suitable large-scale traces available before. With this trace it is also possible to classify and characterize the different types of contacts that occur between vehicles, and to investigate an optimization of scheduling methods for typical contact properties. Moreover, real traces are also needed for the development of a routing algorithm that is optimized for public transport, since the special mobility characteristics have to be analyzed. Further, a reliable simulation-based evaluation is not possible with synthetic traces or models that are more or less building on assumptions instead of real vehicle behavior.

1.1 Objectives and Approach

In this thesis the feasibility and performance of a DTN-based communication system for public transport systems is investigated. First, requirements on the basis of realistic[1] use-cases are analyzed. In the next step the channel characteristics between two public transport vehicles is experimentally evaluated in order to show that the envisioned system is feasible with current wireless technologies. Moreover, this step yields realistic communication parameters for later simulations.

[1] The use-cases (and according business models) were developed in cooperation with BBR Verkehrstechnik GmbH, a supplier of public transport technology.

The next objective is to analyze the node mobility in public transport networks. Therefore, mobility traces of such systems are required. The only existing large-scale trace lacks important metadata and is unsuitable for our purposes. For this reason we recorded a new trace of the public transport system of Chicago, which is one of the world's largest public transport systems. This trace contains additional meta-data on public transport operations. We analyze and compare the existing trace with our own new trace, and use the meta-data of our trace to characterize public transport mobility. In the following step the experimentally determined parameters such as radio range and channel capacity are combined with the mobility traces for a simulation-based analysis of contact characteristics.

Knowing the contact characteristics and mobility properties now allows to derive requirements to a routing algorithm. In the next step this algorithm is designed and implemented. Finally, its performance is evaluated and compared to previously proposed algorithms.

1.2 Scope

The focus of this work is on mobility characteristics and DTN routing in urban public transport systems with vehicles operating on surface infrastructure. Buses and trolleys running on roads or rails are the most common vehicles in these systems. Vehicles move on lines along stops or stations, with arrival and departure times defined by timetables. The core issues that are addressed in this thesis are:

- Identification of use-cases and resulting requirements for a delay tolerant public transport communication system.
- Analysis of public transport mobility with real-world, large-scale traces, as well as the trace acquisition, processing and integration into a simulation environment.
- Experimental investigation of 802.11-based communication performance in public transport scenarios, in order to prove the feasibility of such a communication system and to gain parameters for realistic simulations.
- Classification of situations in which contacts between vehicles occur, as well as the influence of radio range on the statistical distribution of contact duration.
- Proposal and evaluation of a routing algorithm that makes use of public transport characteristics and special properties that were investigated in the above analysis.

The conducted research aims at gaining holistic knowledge and data (e.g. traces, parameters) for the design and realistic simulation-based evaluation of DTN routing in public transport. Thereby, several other areas of research are touched but are out of scope for this work. These are in particular:

- Determination of position data accuracy of existing traces and of the location systems used for our own traces. Due to the Global Position System's intrinsic limits there are deviations that mainly depend on signal reception and receiver quality.

- A real-world implementation of the discussed communication system. This work has a strong focus on feasibility, practical aspects and real-world applicability, but does not go beyond a simulation-based evaluation. However, we deployed a partial and prototypical real-world implementation of the described communication system in the OpTraCom[2] project.

1.3 Contributions

This thesis makes several contributions, such as a detailed analysis of real-world mobility traces, as well as the experimental evaluation of vehicular communication characteristics in public transport systems. Moreover, the conducted research results in the design and evaluation of a novel approach to delay tolerant routing in such systems. In particular the contributions comprise:

- **Trace acquisition and mobility analysis:** The first contribution of this thesis is an analysis of node mobility in public transport networks. For this purpose real-world large-scale mobility traces of such networks are required. Since previously available traces lack important meta-data on the operation of such networks, a novel large-scale trace of Chicago's public transport system was acquired. This new trace is analyzed and compared to existing traces. We report on several unexpected properties and give a quantitative description of the trace properties.

- **Experimental determination of vehicular communication characteristics:** Second, the node-to-node communication characteristics are investigated. We present the results of measurements in real-world public transport scenarios and show the feasibility of 802.11-based DTN communication in public transport. Moreover, our measurement results such as range and data rate are a novel contribution to more realistic parametrization of simulators.

- **DTN contact classification and analysis**: Building on the mobility trace and the communication characteristics the different types of contact in a vehicular DTN are classified. Moreover, their statistical distribution in dependence of radio range is determined. As a result, we contribute novel insights on the situation specific prediction of contact duration and capacity.

- **Proposal and evaluation of a novel DTN routing for public transport:** With our new understanding of public transport mobility we propose a novel routing that makes use of these characteristics. The comparative evaluation that is based both on synthetic and real world traces shows that our routing is both robust and provides a better performance than several previously proposed routing algorithms.

[2] http://www.optracom.de

1.4 Outline

This thesis is organized in 6 chapters. The core contributions are described in chapters “3. Communication Characteristics and Mobility Analysis”, “4. Contact Analysis” and “5. Routing in Public Transport”. Chapters 4 and 5 are building on the results of the previous chapters. In detail this work is structured as follows:

- **Chapter 2** motivates the development of a disruption tolerant communication system in public transport. Besides describing use-cases and advantages, it also provides information on the requirements and prerequisites. Moreover, similar projects and related work are presented.
- **Chapter 3** focuses on suitable license exempt wireless communication technologies and the influence of mobility on communication. First, several real-world measurements of standard WLAN in vehicular scenarios are conducted to determine communication performance and suitability. Second, we give an overview of existing mobility models and traces for simulation of public transport systems. Moreover, we present the acquisition of our own large-scale trace. This trace is analyzed and compared to previously recorded traces.
- **Chapter 4** deals with communication contacts between vehicular network nodes and their influence on the capacity of the network. For this purpose different types of contacts are identified and analyzed. We also investigate the impact of radio range on the statistical distribution of contact durations. Moreover, several scheduling strategies are presented and evaluated.
- **Chapter 5** presents the design and architecture of our proposed routing scheme for vehicular public transport networks, which makes use of the special properties of these networks. After introducing requirements and data models we describe the design and path selection details. Then the router is evaluated and compared to several existing algorithms. Several simulation scenarios with synthetic and real mobility traces are used for this purpose. Moreover, the effect of disturbances and delays is investigated.
- **Chapter 6** concludes the thesis with a summary of the results and possible areas of future work.

Chapter 2

Motivation and Related Work

This chapter motivates a DTN-based communication system for public transport by introducing the applications of such a system. Thereby, similar work and projects are presented, and the characteristics and requirements are analyzed. Moreover, an overview of related work in the area of DTN routing is given.

2.1 DTN Applications in Public Transport

There are several promising applications of DTN in public transport. This section starts with use-cases that are beneficial to the operator of a public transport network. These applications and their characteristics are based on experiences of companies working in this field [1]. Then a more general application is described that uses public transport vehicles to monitor environmental data and DTN technology as a suitable communication channel. From an economical point of view this mix of applications establishes synergies and allows for a stable business model.

2.1.1 Operator Machine-to-Machine Use Cases

2.1.1.1 On-board Monitoring

In this first use case, on-board sensor data is acquired by public transport vehicles and sent via the DTN to a common sink, i.e. a DTN Internet gateway and from there to the control center of the public transport operator. The monitoring data is usually needed both for long-term capacity planing as well as for short-term dispatching of additional vehicles if they are overcrowded or out of order. The data is also important to schedule regular vehicle service and to trigger preventive maintenance.

Some examples of relevant monitoring data are:

- vehicle's remaining passenger capacity, measured by passenger counting systems at the doors
- vehicle's technical health, e.g. the engine temperature obtained from the vehicle bus
- floating object data, e.g. traffic conditions and average speeds at rush hour

Characteristics Network traffic caused by monitoring applications has specific characteristics. In a DTN scenario, data from most sensors is stored for a certain interval and then transmitted in bulk in a single bundle. Only sensor readings of unusual or critical values which must be delivered urgently, trigger an immediate transmission. Therefore, monitoring applications cause a relative constant bundle rate. The amount of data is low and in the order of some kilobytes per sample since it only comprises several sensor values and timestamps. The requirements regarding the delivery delay of a bundle are low as long as no critical thresholds are exceeded. However, for critical events where a short-time treatment is needed, an expedited bundle should be delivered to the control center within a few minutes.

2.1.1.2 On-board Passenger Information Systems

On-board passenger information can be implemented in various ways and often includes voice announcements and display panels. A very common application is to inform passengers about the name of the next stop. Further, there are displays showing advertisements and news. Moreover, special events, changes (e.g. changed routes or timetables) and disturbances of the public transport network (e.g. traffic jams, reroutes, breakdowns) are communicated.

The aim of this use case is to update such information systems via the DTN. Information displayed or announced is often adapted to specific routes, vehicles or locations. Therefore, a simple broadcast is not a suitable approach to distribute such receiver-individual information.

Characteristics Since updates may contain voice announcements and multimedia content, they may have a size in the order of 10 to 100 megabytes. The network load can be characterized as bursts because updates of content are most likely happening at irregular intervals. Depending on the type of content, the delivery delay should not exceed 20-60 minutes for updates of news, and a few hours for advertisements.

2.1.1.3 Information Distribution to Bus Stops

Bus stops are usually equipped with printed information about lines that service this stop as well as the departure times of these lines. The distribution of printed information is a complex and expensive process. If there is a change in schedules or routes, information displays at each affected stop have to be swapped almost simultaneously over night. In a large city like Chicago with almost 12000 bus stops this is a logistical challenge. In the near future, paper information will be replaced with displays, e.g., based on electronic paper [2]. These bistable liquid crystal displays are very energy efficient so that they can be solar powered. Moreover, energy efficient short range radio technologies such as IEEE 802.15.4 will be integrated in these displays.

In this use case, the electronic displays at bus stops are updated via DTN transmissions from buses passing by. Since public transport network (PTN) operators want to use the corporate design for this information, it can be assumed that the displays will accept data in a standard document format such as PDF.

Characteristics The updates are transmitted from a central DTN-Internet-gateway to the vehicles and from there to the electronic displays. A route or timetable change usually involves multiple bus lines, and therefore a large number of bus stops. Thus, the network load caused by this use case is characterized by a significant burst. The expected update size (based on the assumption of a standard document format) is in the order of several ten to several hundred kilobytes per stop.

Changes to routes or timetables are generally planned long before the timetables are distributed to the stops. An advantage of electronic information is that the change of the displayed information can be triggered by a timer. So the new information can be transmitted to the electronic displays hours or days before it becomes effective. Then, at a defined time, all displays at all stops start to show the new information. Therefore, the requirements regarding the delivery delay of the data to the stations are relatively low, since a delay of several hours (or even days) is acceptable.

2.1.2 Use Case "Air Quality Monitoring"

The measurement of environmental data in city areas has become an important issue to municipalities due to several national, European and even international climate directives. However, fixed measurement stations are inflexible, cost-intensive and limited to monitoring environmental data in distinct key areas only. Large-scale data collection with a high spatial resolution requires mobile measurements, which are an active area of research but still offer many challenges.

Air pollutants caused by vehicular and industrial emissions have a negative effect on human health. Especially in agglomeration areas air pollution may have a direct impact on mortality [3, 4]. In combination with certain geographical and meteorological conditions, air pollution causes dangerous smog. In 1952, the Great Smog of London caused 12.000 premature deaths [5]. This cataclysmic event initiated emission research, government regulations on air pollution and monitoring of air quality. Today, most developed countries have implemented effective processes to control air pollution. These comprise long-term measures such as land use planning, regulation, and tax incentives for clean air technology, as well as short-term measures like traffic management and road space rationing. Nevertheless, it is not uncommon that legal threshold limits are exceeded.

Air pollution control requires constant air quality monitoring in order to develop and evaluate pollution countermeasures. The state-of-the-art approach to monitoring is the sparse deployment of fixed measurement stations. However, it is more desirable to acquire data with a high spatial resolution, but that would require a large number of well distributed measurement stations, and therefore be very expensive. A more cost-effective approach is to perform mobile measurements. By constantly moving the monitoring equipment, a high spatial resolution can be gained with just a few measurement devices. In the past this approach was not feasible due to the physical dimensions of the monitoring equipment and due to the lack of appropriate mobile communication infrastructure. Nowadays, advances in sensor design, mobile computing, and pervasive mobile communications make it possible to implement measurement devices that can be mounted to vehicles or even be integrated into smartphones.

This section gives an overview of the multidisciplinary field of mobile air pollution monitoring. Several different approaches and research projects are presented.

2.1.2.1 Applications and Requirements

Air pollution monitoring is implemented for various reasons with different requirements. The most important application is probably to check if legal limits are exceeded, so that countermeasures can be initiated and the population can be warned if pollutant values reach a health critical level. Often laws and directives make pollution monitoring mandatory [6, 7, 8, 9]. Therefore, the measurement equipment and process used for this purpose has to be standardized, calibrated and certified for legally valid results.

Monitoring is also necessary to evaluate measures against pollution. For this purpose, it is important to record weather conditions, since wind, humidity and precipitation have a significant influence on the distribution and accumulation of air pollutants. Therefore, results of different points of time are only comparable at similar weather conditions. Another application is land-use planning and traffic planning with the aim to reduce pollution in areas in which the levels are regularly exceeded. This requires that local pollution and its causes are examined, and that the effectiveness of the measures has to be evaluated afterwards. Usually several measurement campaigns with portable equipment are required to obtain a spatial resolution high enough to assess local effects. An application with similar evaluation requirements is dynamic road traffic control. The basic idea is to develop traffic control strategies that help to avoid exceeding limits, such as traffic redirection or reduction in critical areas, ban of certain vehicles from low emission zones[1], or emergency measures like road space provisioning and even general driving bans.

An important application for the citizens is being informed if pollutant values are getting critical. Especially very young and elderly people, but also patients suffering from heart or breathing conditions are more sensitive to pollutants. But also healthy people should avoid to work out at places or at times of high pollution. Therefore, such a citizen information system needs to distribute localized and personalized information. Existing systems use websites with interactive maps and short message services[2]. Future systems will likely be implemented as smartphone apps.

2.1.2.2 Data Acquisition Approaches

There are several approaches to the acquisition of measurement data on air pollution. Currently, the measurement of air pollution is realized on the basis of few fixed measurement stations. The concentration of harmful substances in the air is predicted based on the measured data in combination with complex mathematical models. Nevertheless, this predictive approach goes along with inaccuracies and is not very well suited for realizing a sustainable traffic management in order to avoid exceeding prescriptive values and a complete information of the population. Deploying additional stations to obtain a higher spatial resolution is not really an alternative, since this would require high investments.

[1] http://www.tfl.gov.uk/roadusers/lez/default.aspx

[2] http://airtext.info/

But instead of using only a few fixed measurement stations, a mobile measurement network can be used to gather area-wide measurements of the air pollution. Both the fixed monitoring stations and several approaches to mobile measurement are described in the following.

Figure 2.1: Aerosol monitoring station. Image by Breitfuss Messtechnik GmbH, Germany.

Monitoring Stations Monitoring stations are widely used and well proven. They usually comprise some kind of container or shelter with a size of several cubic meters, which holds racks of measurement devices and a supply of calibration gases and other consumables. Air inlets feed the sample gas from the outside to the measurement devices for NO, NO_x, O_3, CO, CO_2 and particulate matter. For the particular matter usually only the amount for different classes of particular sizes is measured. The composition of the particulate matter is then determined off-site, by chemical analysis of air filters which collected particular matter at the measurement station. A local computer (like in the monitoring station shown in figure 2.3) collects and processes the measurement results, which are then transmitted to a remote central database. Moreover, a measurement station also comprises meteorological devices, since weather has a significant influence on the concentration of emissions. Usually, there are at least devices for wind speed and direction (anemometer and a wind vane), as well as humidity and precipitation (hygrometer and a rain gauge), and temperature. Figures 2.1 and 2.2 show monitoring stations with visible weather sensors.

Figure 2.2: Air quality monitoring at an airport. Image by Breitfuss Messtechnik GmbH, Germany.

For economical reasons, monitoring stations are very sparsely distributed. It is often criticized that the local results are not representative and therefore not necessarily correlate with the pollution that citizens are exposed to in daily life. However, it is possible to use numerical models to extrapolate air pollution for other locations based on meteorological data, although the accurateness of these models is hard to verify. Nevertheless, monitoring stations are a proven, reliable technology and the state-of-the-art method to gauge pollutants.

Mobile Devices and Crowdsourcing The idea behind using mobile devices (such as smartphones or stand-alone mobile sensors) is to make use of peoples' daily mobility. The main advantage is that this mobility comes for free (people move anyway) and that measurements are performed where people are, therefore the results correlate to real pollutant exposition (what people breathe in). Furthermore, a large number of well distributed measurement values can be gained if there are multiple users carrying sensor-enabled devices. However, there is also a strong challenge: People have to be

Figure 2.3: Measurement equipment within a monitoring station. Image by Breitfuss Messtechnik GmbH, Germany.

motivated to carry these mobile devices. Moreover, performing a measurement may require user interaction. Therefore, users have to be offered some kind of incentive for participating in the crowdsourcing of air pollution measurements. Another challenge is the development of sensors that are small enough for the integration into smartphones. This is already possible for some pollutants like CO, CO_2 and NO_2, but sensors for some other pollutants (e.g. particulate matter) are a big challenge. There are several approaches to mobile measurements, which range from off-the-shelf smartphones to dedicated mobile sensors.

In [10] a very notable approach to measuring particulate matter concentration is proposed. Its strength is that it uses off-the-shelf smartphones without any additional hardware. The neat trick is using the camera: a picture of the sky is sufficient to gauge atmospheric visibility, which correlates with particulate matter concentration [11]. Determining the exact position and angle at which the image is taken is quite easy, since nowadays almost all off-the-shelf smartphones feature GPS-receivers as well as a compass and accelerometers. This data and the image is transmitted to a central server, which performs most of the processing tasks. The remaining challenges of this approach are that it requires a radiometric calibration of the camera and that the model that derives pollution from visibility has not yet been verified. However, the proof-of-concept evaluation shows very promising results.

In the CamMobSens[3] project dedicated devices were used for portable measurements of CO, NO and NO_2. These devices connect via bluetooth to the users mobile phone, in order to upload the measurement results to a database. The developed visualization is based on Google Earth. Another project using dedicated devices is Common Sense[4][12], in which a cellphone-sized device with sensors for CO, O_3 and NO_x was developed. At the moment a deployment at a community action group is prepared.

Vehicle mounted Devices Measurement equipment can also be mounted on vehicles to obtain a high spatial resolution of the measurements. The advantage over user carried mobile devices is that there are not so many restrictions regarding physical dimension and energy consumption. However, it is not possible to measure an individual user's exposition. On the other hand, vehicles move at higher speeds and larger distances, thereby enabling a much higher coverage area per device. Moreover, it is possible to chose special purpose vehicles with high usage rates that operate in the target areas, such as sweep vehicles [13] or public transport vehicles [14]. Several projects that follow this approach are introduced in the following sections.

Info-Regio The Info-Regio [15] project has been carried out in the run-up to the world fair EXPO 2000. Its aim was to develop a sustainable and efficient traffic management system. Although this project finished ten years ago, it is notable for two reasons. Firstly, because it has been shown that the approach of monitoring environmental data on moving vehicles is feasible and a sensor prototype was developed. And secondly, because the project was not fully realized due to problems with the communication infrastructure. Therefore, this project is a nice example showing that multidisciplinary research and engineering efforts are required for the realization of mobile air pollution monitoring.

Being the first project which successfully developed and deployed a suitable bus mounted sensor was one of the projects' achievements. For this purpose an air inlet was installed at the roof of a public transport bus. From the inlet the sample air was pumped through a hose to a carbon monoxide measurement device. Two fixed measurement stations were used for comparison and showed the feasibility of the mobile measurements in an extensive evaluation.

An unsolved problem was the communication infrastructure. It was planed to use the existing private mobile radio of the local public transport carrier. However, the project members realized very quickly that the bandwidth of this private mobile radio was not sufficient to transfer the additional measurement information. At the end of the project, two options for realizing the data exchange remained. On the one hand, the measured data might be transmitted via cellular radio networks at high operating costs. This possibility was used for the system's prototype, but was not feasible for the planned large and long-term deployment. On the other hand, the project partners proposed to install a separate private mobile radio system dedicated to the exchange of measurement data. However, this solution would have led to high investment costs and was therefore not realized. Although the operating costs for 3G data services have decreased since that

[3] http://www.escience.cam.ac.uk/mobiledata/

[4] http://www.communitysensing.org/

time, the basic problem still remains. The transport operator either has to purchase communication as a service or invest into his own dedicated infrastructure. Installing an infrastructure-less system that is based on on-board units would be a more attractive solution.

AERO-TRAM In the AERO-TRAM[5] project, measurement equipment similar to that of monitoring stations was installed on a tramway in Karlsruhe, Germany. This project focuses on very precise measurements, therefore the equipment choice was relatively uncompromising: the roof-mounted container weights roughly two metric tons and includes a supply unit with consumables such as calibration gases and chemicals. The lab-quality equipment measures NO, NO_x, O_3, CO, CO_2 and particulate matter. A custom, carefully designed aerosol inlet regulates volume current. This unit is mounted in the front of the tramway's roof and avoids that emissions from the tramway are influencing the measurements. Moreover, the constant volume current ensures speed-independent measurement results. The aerosol inlet and the equipment in the containers are shown in figures 2.4 to 2.6.

Figure 2.4: AERO-TRAM with roof mounted container and aerosol inlet.

[5]http://www.aero-tram.kit.edu/

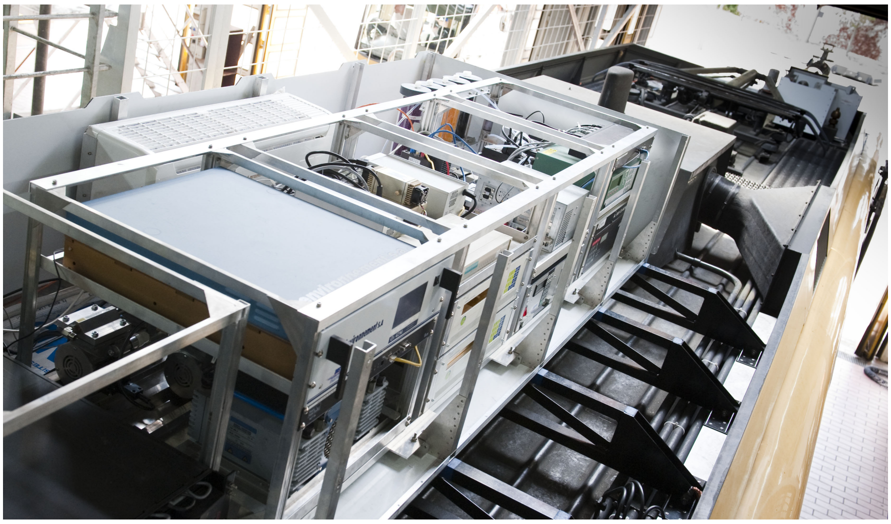

Figure 2.5: Rear container with mobile lab equipment.

The tramway operates on a 30km long route that covers the city center as well as the hinterland. This allows for an analysis of urban effects and the spatial distribution of pollutants. Moreover, models used to extrapolate a spatial pollutant distribution from stationary measurements can be evaluated. The results of mobile measurements are also of great value for evaluating the validity of stationary measurements, since critics often question the representativeness of local pollutant values.

In conclusion AERO-TRAM is to the best of our knowledge the most precise and sophisticated design for mobile air pollution monitoring, but also the largest, heaviest and most expensive solution. However, these drawbacks are unavoidable, because the high precision is required for the purpose of evaluating the effectiveness of stationary measurements and distribution models. [16]

EMMA EMMA [6] (Environmental Monitoring in Metropolitan Areas) [17, 14] builds on previous projects that have shown the feasibility of mobile air pollution monitoring. Its focus is not on the development of sensors and transducers for environmental data, but on implementing a cost efficient, self-organizing and maintenance-free communication infrastructure for large scale deployments. In previous projects like Info-Regio, communication was an unsolved problem. It turned out that there was no technology with both low investments and low operational costs. In consequence a profitable operation was not possible.

[6] http://www.ibr.cs.tu-bs.de/projects/emma/

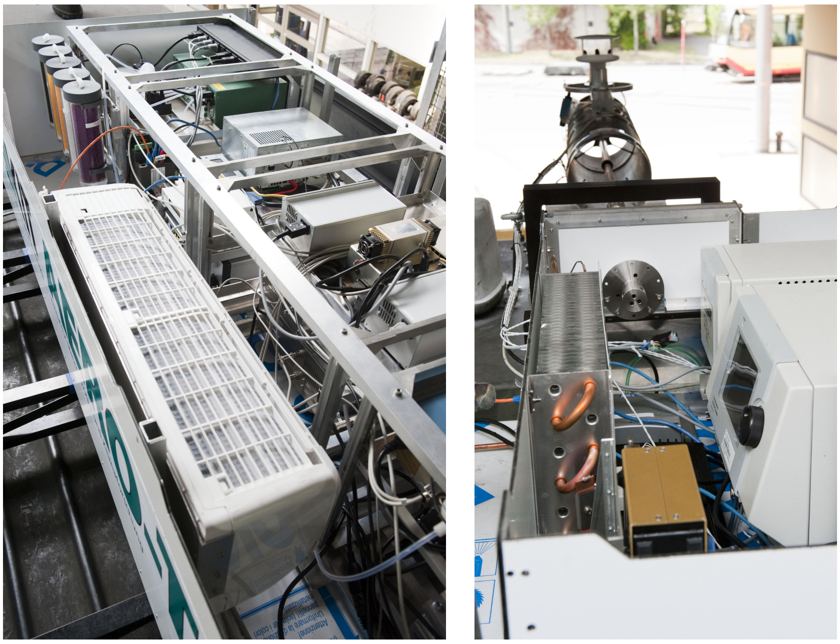

Figure 2.6: Closeup of the rear container and the smaller front container with the aerosol inlet.

In EMMA, a decentralized architecture for environmental monitoring with vehicle mounted sensors using DTN communication techniques was developed. It can be integrated in, e.g., existing public transportation networks by equipping buses with sensor nodes that communicate with each other. The system is able to obtain area-wide measurements that are distributed by an ad-hoc network. Buses provide almost complete coverage of the whole metropolitan area (at least for major roads), so that a relatively small number of measurement devices is sufficient and hardly any data network infrastructure is needed, as the sensor nodes spread their data all by themselves. The whole system is self-organizing and thus requires little to no setup efforts. Whenever a node discovers another node within its transmission range, both start the data exchange process. Beginning with the newest set of measured data, each node sends all stored information to every other node within communication range. This makes EMMA an inexpensive and flexible solution that is easy to implement. In section 2.1.3 a detailed overview of EMMA's architecture is given as an exemplary application that builds on a DTN. This thesis is based on EMMA as a motivating example, but focuses on message dissemination and routing in EMMA and similar systems. Currently, these systems mainly use flooding approaches. There is no doubt that flooding is a feasible and robust solution [18] for networks with low traffic load. However, it also wastes significant amounts of network capacity with replicated messages. In order to implement more general DTNs that are not limited to specific, low-load applications, a more efficient routing strategy is required.

2.1.2.3 Data Transmission, Processing and Presentation

High spatial resolution measurements with mobile devices require a high sampling rate of the moving sensor. Therefore, a large amount of data needs to be transmitted to a central database which stores and processes the results. The amount of data of a single sample is in the order of some hundred bytes. It contains the GPS position, date and time, sensor identifiers, and readings of several sensors. A realistic sampling rate of current sensors is between two seconds for carbon monoxide and six seconds for particulate matter. With regard to the amount of data generated by a single mobile device the requirements on the communication system are not too demanding. Nevertheless, a large-scale deployment with many vehicles puts high demands on the communication system, because wireless communications is a shared medium. Moreover, it is very important that the communication system has low operational costs.

The load on the central database scales with the number of vehicles. For a large-scale (e.g., nation-wide or even planet-wide) deployment it will be necessary to implement a hierarchical architecture, e.g., with databases limited to certain regions. This approach would prevent bottlenecks, but the drawback is that a cross-region database request would suffer from bad performance. However, such requests are relatively rare, since the usual use-case is location-centric.

Pollution alerts and the visual representation of the pollution is important for the information of citizens. The visualization engine's task is to aggregate this data and prepare it in a fashion that allows the key implications to be understood intuitively. As a lot of information in this data is location specific it is especially useful to chart it on a map. That way it is easy to get a global overview but also have location specific details. It is necessary to make this map easily accessible to residents as well as to visitors of a city. This can be realized, e.g., via Internet services, display panels or novel mobile information systems such as apps for smartphones. Thus, people who are sensitive to specific pollutants can easily get informed about the current situation. Such maps are also powerful tools for traffic-flow and land-use planing.

2.1.2.4 Conclusions

There are several promising approaches to mobile air quality monitoring. The basic feasibility of small-sized sensors mounted to vehicles or as stand-alone mobile devices has been shown. However, the demanding requirements of large-scale deployments need a joint effort of multidisciplinary research. In the near future, datasets of first prototypical deployments will be available, and hopefully a common data repository will be established. This data will be a key contribution to the understanding of the complex distribution, aggregation and resolving of pollutants in urban areas. In these areas DTN is a feasible solution for data transmission. Although there are other options, e.g. infrastructure based mobile communications, a DTN approach is able to satisfy all requirements without the need (and investments) for additional infrastructure.

2.1.3 Exemplary Architecture for a DTN-Based Environmental Monitoring System

This section gives a more detailed overview about an exemplary application of vehicular DTNs. We developed a distributed environmental monitoring network called Environmental Monitoring in Metropolitan Areas (EMMA). This architecture is based on the delay tolerant networking approach and can be integrated into existing public transportation networks. Buses or other vehicles can be equipped with sensor nodes that gather data and forward messages. In order to evaluate the basic ideas of this project we performed a series of real-world experiments. Besides analyzing the behavior of 802.11 based WLAN between moving vehicles in a controlled environment, we also evaluated the communication performance in urban environments. Moreover, we examined the qualification of a DTN implementation for spreading measurement results throughout the network. The suitability of EMMA's architecture has been successfully demonstrated by these experiments.

2.1.3.1 Requirements

The EMMA project (Environmental Monitoring in Metropolitan Areas) proposes a communication architecture that can be integrated into existing public transportation networks to allow area wide pollution measurements. The sensor nodes can be installed in e.g. buses or trams where they continuously measure the environment. The collected data has to be centrally aggregated in order to analyze the air quality in the city or in specific districts. A simple approach would be to use an available infrastructure network such as GPRS or UMTS. However, due to the permanent data exchange and, hence, the incurred high costs, the total operational costs would likely prohibit the introduction of such a system. We decided to base our system on a decentralized approach using a self-organizing ad hoc network. Sensor nodes may share information whenever two buses meet and thereby spread information throughout the whole public transport network. It is possible to forward the measured data to a central server or external network via one or more gateways which may be installed at major bus and train stops.

This way it is possible to establish a cost-efficient dynamic traffic management system that considers up-to-date information on current pollution instead of relying on mathematical models that extrapolate from the data measured by only a few fixed measurement stations. In addition, each bus may be equipped with a passenger information system that combines the measurements of several buses and thus provides information on the local situation to people that might be sensitive to specific pollutants. Of course, the distributed architecture is not limited to exchanging measurement data but may also be used for spreading announcements or other information that can be used in passenger information systems.

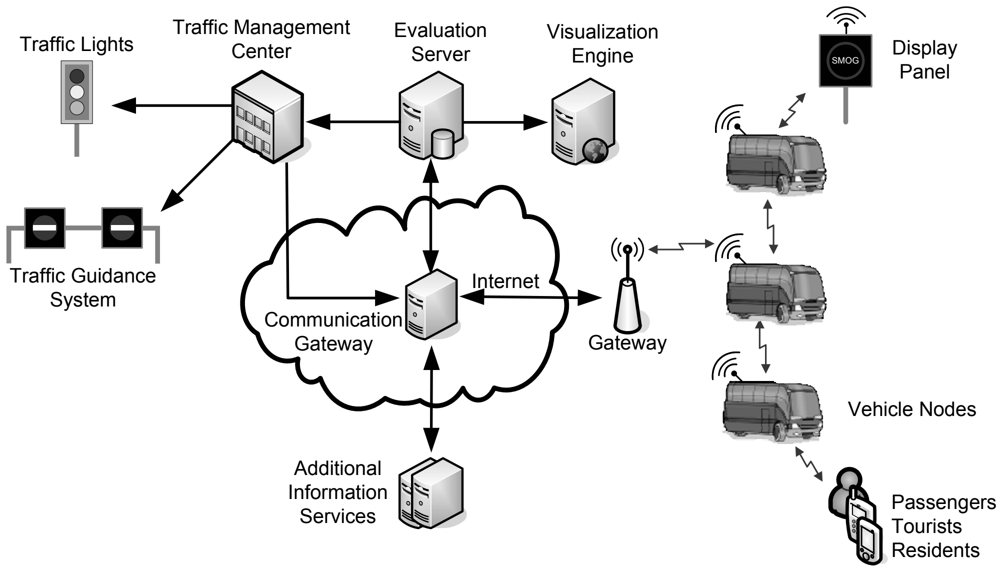

Figure 2.7: Basic architecture of EMMA.

2.1.3.2 System Architecture

Various parameters affect the dissemination time of datasets throughout the network, such as the area covered by the public transport network and the frequency of buses serving a given route. Figure 2.7 shows EMMA's architectural elements, which can be structured in two groups: the elements forming the delay tolerant network (mainly DTN nodes in different configurations) and the elements connected by a regular WAN (traffic management and control). Each node comprises a wireless network interface, a DTN service and some data storage capacity. Sensor nodes are additionally equipped with sensors (e.g. NO_x), a GPS receiver and an optional vehicle interface. Plain DTN-nodes act as mobile relays to speed up data distribution.

Besides vehicle mounted nodes, there may also be two types of stationary DTN nodes: gateways and smart display panels. A gateway connects the DTN to the traffic management WAN and therefore is best installed at a central location, e.g. a main intersection. The gateway receives messages (so-called bundles) from the DTN and forwards the measurement results to the evaluation server via a WAN connection. It also forwards messages (e.g. control messages for traffic management devices or information displays) from the WAN to the DTN. Smart display panels are the second type of stationary nodes. These displays show information on current pollutant concentrations. The measurement results are gathered from passing vehicles and computed autonomously by the panel. Further, the display may also receive messages from a control center (via the DTN) in order to show e.g. traffic information. Displays also act as relays, speeding up the bundle distribution process at no additional cost.

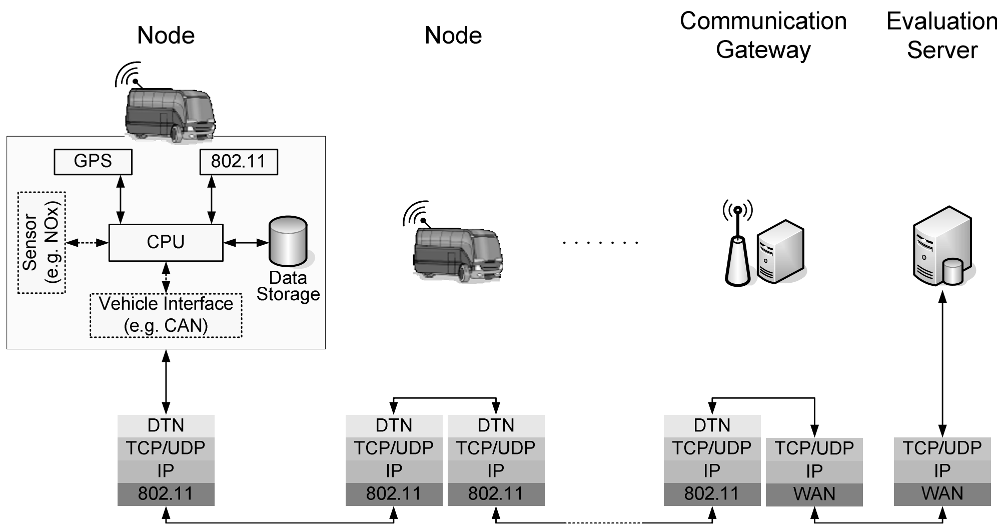

Figure 2.8: The EMMA protocol stack.

EMMA's WAN-side elements are the communication gateway, the central evaluation server (CES) and the visualization engine. The communication gateway connects EMMA's various subsystems with each other and the Internet, and provides an interface for the integration of additional information services (e.g. electronic tourist guides). It also translates DTN bundles to regular TCP streams and vice versa (Figure 2.8). The central evaluation server is the final sink of all measurement data originating from the DTN. It comprises a database and a set of rules to trigger events based on the computed pollutant concentration of certain areas. An example for such an event is sending a signal to the traffic management center if the particulate matter threshold is exceeded. The traffic management center, already installed in most major towns, controls the traffic flow with traffic lights and guidance systems. On receiving a signal from the evaluation server, the traffic management center can either reduce the traffic load in polluted areas or even entirely close down certain roads for specific vehicles (e.g. trucks or cars without catalytic converters). The visualization engine generates a human readable city map with an overlay of the pollutant concentration from the evaluation server's database. A web-interface (using Google Maps [19]) provides the map along with area-specific information to citizens. Furthermore, such maps are powerful tools for traffic-flow and land-use planning.

Figure 2.9: Vehicular measurement and communication device.

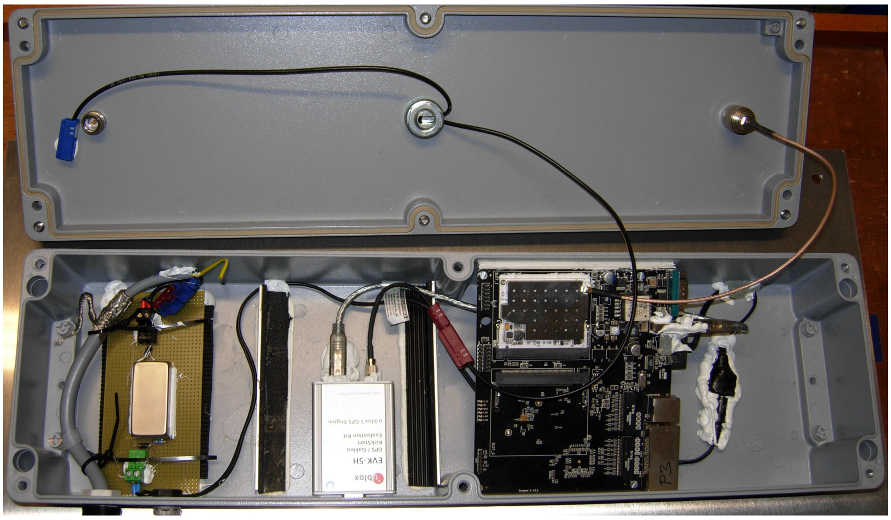

Figure 2.10: Internal view of vehicular communication device, with power supply and surge protection circuit (left), GPS receiver (middle) and embedded system with radio and mass storage (right).

2.1.3.3 Similar Projects

The DTN approach used by EMMA is based on a general architecture for delay tolerant communication proposed in [20]. Previous work in this area deals with various routing mechanisms for data delivery or different applications of DTNs. The UMassDieselNet project [21] is closely related to EMMA. In this project, 40 buses were fitted with off-the-shelf communication hardware for evaluating the performance of bus-to-bus and bus-to-infrastructure communication. The buses send location information as well as measured data to a central server whenever Internet access is available. Although several aspects of DieselNet are comparable to EMMA, it is not a distributed information system where measured data is spread through the network. Since both projects have different application scenarios, requirements on the communication performance differ as well. An interesting part of the DieselNet infrastructure are throwboxes [22] that are intended to increase the network connectivity and relay data. The concept of relays may also be helpful for EMMA in order to ensure the data exchange between bus lines that meet

Figure 2.11: Communication device installed on the roof of a public transport vehicle.

infrequently. However, the design requirements for EMMA relays are different. DieselNet throwboxes run on batteries, and therefore are designed with a focus on low-power consumption. For EMMA, a long radio range and a processor that is powerful enough for high data rates is more suitable. Moreover, mains power supplies are available at most bus stops, and is preferable over rechargeable batteries which have a limited lifetime.

Our work is also related to ferry-based networks [23, 24, 25]. This research area aims at connecting nodes in sparse networks by using robots, buses or other vehicles for asynchronous data transport. In the DakNet project [26], such networks are used to provide Internet access to remote villages in India and Cambodia. However, the requirements of ferry-based networks differ from EMMA since data is only exchanged at specific access points and delay is limited to the travel time between a village and the Internet gateway. Other related work in the area of DTN include [27, 28]. The communication performance of vehicle-to-infrastructure communication has been studied in [29, 30, 31]. In contrast to our work, they do not analyze characteristics like signal strength and data rate over the distance of two moving vehicles. Additionally, the performance of DTN bundle exchange in urban environments has not been evaluated in previous work.

2.1.3.4 Proof-of-Concept Implementation

Figures 2.9 to 2.13 show our proof-of-concept hardware implementation of the different types of DTN nodes that are used in our architecture. The communication device with antennas for WLAN and GPS is shown in figure 2.9. The enclosure in figure 2.10 also

Figure 2.12: Public transport vehicle equipped with communication device.

contains an embedded Linux platform, power management, sensors, GPS receiver and high range WLAN interfaces. In figure 2.11 the advantages of the modular mounting are shown: the device is easily integrated because it is designed to fit the modular roof construction of public transport vehicles. Figure 2.12 shows the same vehicle in operation, the communication device is located near to the middle in front of the pantograph. Moreover, we have developed solar-powered DTN relays [32] in order to boost the communication performance. Although these relays could be considered as architectural "infrastructure elements", they do not require any wired infrastructure or cabling themselves. These energy-autonomous nodes are completely wireless and therefore extremely easy to deploy. Moreover, the installation is very economic because the main deployment cost (and hassle) is caused by cabling work. One of our solar-powered installations is shown in figure 2.14.

The purpose of EMMA is not only limited to implementing a distributed measurement system. The Internet gateways may also be used to offer several information services. A passenger information system using the EMMA architecture may, e.g., include the distribution of news or special announcements of current events in the city. When vehicle-to-vehicle communication is introduced to the mass market, equipped vehicles may also benefit from the information services provided by EMMA.

With the EMMA architecture we have laid the conceptual foundations of a DTN-based communication system for public transport. Before the system can be implemented in a large-scale public transport network, there are several investigations required to ensure a satisfactory performance. Especially the influence of real world mobility on communication characteristics and the implications for the design of a suitable routing algorithm have to be analyzed.

Figure 2.13: Communication gateway, connecting the DTN to the Internet.

2.2 Approaches to DTN-Routing

In this thesis relevant existing routing approaches are divided into three categories: i) Epidemic routing and variants of flooding, ii) probabilistic approaches and iii) schemes that use context information. This categorization is intended for the readers orientation, since there is a large number of diverse approaches. It should be noted that several different categorizations have been used in previous work [33, 34].

2.2.1 Forwarding, Flooding and Variants

Several generic DTN routing protocols have been proposed that can be applied to urban public transport networks. FirstContact routing [35] is following a hot-potato approach and forwards a message to the next available node until the destination is reached. A random node is selected if more than one node is in radio range. FirstContact requires only a minimal message buffer on the nodes but causes a relatively high delay and network load.

Epidemic routing [18] is a simple but effective routing method based on flooding. Messages are forwarded to all neighbors if the receiver is not directly reachable by the sender. Each neighbor maintains a buffer for flooded messages. Upon meeting another node a summary vector is exchanged to prevent unnecessary transfers of messages that both nodes have already buffered. Messages are discarded in a first-in-first-out manner if the buffer's capacity runs low. Message lifetime is limited by a hop counter. Epidemic Routing achieves a low latency but causes a high network load.

Spray and Wait [36] tries to reduce the overhead of message copies by prohibiting non-source nodes to create message copies. Only the source node creates a limited number of L copies. The routing is divided into two phases: In the spray-phase the nodes spray and forward messages and in the wait phase they wait for a direct transmission to the destination node. Whenever a node discovers another node which has not yet buffered the message, the source node starts with the spreading and forwards one or L/2 message copies to the new neighbor. The spreading is stopped as soon as nodes keep only one copy. Afterwards both nodes change to the wait phase and stand by for a direct delivery.

Spray and Wait routing spreads messages quickly to the first nodes contacted. This can be problematic if exactly those nodes cover only a limited area. In this case far destinations may not be reached. Therefore a follow-up approach has been proposed: Spray and Focus[37] generates L tokens for every new message. These determine the number of possible copies and forwardings. Similar to Spray and Wait routing, the nodes forward L/2 tokens when the message is known on only one side. When all tokens have been used, the node goes into the Focus phase. In contrast to the Spray and Wait routing, messages can still be relayed. It calculates a utility function which decides about the transfer. For example this utility function could be derived from the expected duration of a contact.

2.2.2 Probabilistic Approaches

PROPHET [38] is a probabilistic routing protocol using a delivery prediction based on historic encounters. The forwarding strategy is similar to epidemic routing, but the resource utilization is reduced by forwarding only a few message duplicates to nodes with a high delivery predictability. Like in epidemic routing the nodes exchange summary vectors on each encounter. These vectors contain the identifiers of messages in the nodes' buffers as well as updated predictability information. PROPHET can be applied to routing in public transport systems but does not exploit their specific characteristics.

In comparison to epidemic routing the better resource utilization of PROPHET achieves a higher message delivery rate if the buffer capacity is low on a significant number of nodes. In this case old messages have to be discarded in order to store new messages, so that some cannot reach the recipient because all copies are discarded by nodes with full buffers. On the other hand epidemic routing causes a lower delivery latency if the buffer is sufficient on all nodes. The nodes' mobility pattern has a heavy influence on the success of PROPHET. Its performance is generically better in scenarios with repetitive mobility. Nevertheless, the evaluation of Lindgren [38] shows a marginal advantage over epidemic routing in a random waypoint scenario.

MaxProp Routing [39] is optimized for DTNs with fast moving nodes. It is based on flooding but prioritizes messages with the lowest costs caused by forwarding the message to its destination. The costs reflect the delivery probability. In addition, each node maintains vectors with an estimation of the probability of meeting every other node. At each contact, the vectors are updated and re-normalized. Moreover, the nodes exchange their vectors. Then the potential paths to the destination node is calculated. The costs result from the encounter probabilities. Like PROPHET, MaxProp can be applied to public transportation systems. However, both routing algorithms rely on historic encounters and do not proactively exploit a public transport system's properties such as timetables and lines.

Routing in Cyclic Mobispace (RCM) [40] is a model of cyclic repeating node movement based on a multidimensional euclidean space [41, 42]. This space maps the spatial probability that a mobile node is located at each possible position. RCM is a long-term routing approach. In the described public transport scenario, information on contacts between nodes is collected for several weeks, and then used for routing decisions for several years. During operation, changes (e.g. additional or removed vehicles) are distributed as incremental updates. The overhead caused by these updates is relatively low, since such changes do not occur often. A long-term metric called Expected Minimum Delay (EMD) is defined to maximize delivery ratios without implementing bundle duplication. For the evaluation UMassDieselNet traces are used. The underlying public transport network has a daily non-deterministic vehicle-to-route assignment, but the authors tried to extract cyclic contracts from the trace. Furthermore, "NUS student contract traces" and two synthetic public transport traces of Madrid and Miami are used. In the evaluation the metrics delivery ratio, delay and forwarding hop-count are studied. RCM is compared to Epidemic, Spray and Wait, Spray and Focus and MaxProp. RCM is only outperformed by the delay of Epidemic.

Multi-copy Forwarding in Cyclic MobiSpace (MFC) [43] is an extension to RCM and introduces bundle duplication. In contrast to single-copy-forwarding, several bundle duplicates may simultaneously exist in the network. In general, duplication may increase delivery ratios in the case of failed contacts or if a few of many nodes are overloaded. For the evaluation a certain number of contacts is eliminated from the UMassDieselNet traces, so that a defined uncertainty is induced to the routing information. As to expect, MFC achieves a higher delivery ratio than RCM under these conditions.

Trajectory Prediction DTN Routing (TPDR) [44] is an approach to determine the probability of contacts per time unit based on a Markov model. It was developed for networks of "small and medium size", as an example the authors give a DTN of a few autonomous underwater vehicles. The approach assumes that the nodes can determine their positions and that a prediction of trajectories is possible. Moreover, the nodes must have enough knowledge on the mobility pattern of all other nodes in order to predict the start and end of contacts. The authors claim that these conditions are fulfilled in centrally or autonomously coordinated systems, such as trains and mass transit, but do not clarify details. Moreover, it is assumed that nodes move with constant velocity on predetermined trajectories, but may halt their movement in the case of "detection events". It seems that applications, assumptions and the derived model are not entirely consistent. The model is based on a two dimensional XY-grid in which each node is located on a certain tile with a known probability distribution. It stays there for a certain duration ("state holding time") and then moves to another tile, thereby changing to another state. Therefore, it is possible to determine the probability that a node will be at a certain tile at a given time, since the trajectories are known. The nodes use a neighbor map to store the contact profiles of other nodes. At a contact, bundles are forwarded to another node if its contact probability with the destination node is higher. The evaluation is based on a Matlab model and shows that the delay of TPDR is higher than the delay of epidemic, but lower than Direct Delivery (which forwards messages only at a directly to the receiver node). However, the delivery ratio of Direct Delivery is significantly lower and epidemic is only slightly better than TPDR.

2.2.3 Context Information

GeoDTN+Nav [45] is a hybrid approach for VANETS, that falls back to a Store-Carry-Forward mode if the network is partitioned. For this purpose the authors propose a standardized Virtual Navigation Interface (VNI) which is used by the vehicles to exchange information on their mobility patterns and properties. Vehicles are assigned to categories such as “Deterministic Route”, “Deterministic Destination”, “Probabilistic Route/Destination” and “Unknown”. Routing decisions are made based on the information provided by the VNI, which in turn gains this information from the vehicle’s navigation system. There are three different routing modes: the two classic approaches to VANET routing called “Restricted greedy forwarding” as in GPCR and “Perimeter forwarding” as in VCLCR. The third mode is DTN forwarding. A suitable mode is selected with a score function. First the probability of a partitioning of the network is estimated based on hopcounts. Then the information from the VNI is used to gauge delivery quality and direction quality of neighbor nodes. Data is forwarded in DTN mode if the score is above a threshold value.

GeOpps [46] uses geo information obtained from a vehicle’s navigation system. Destination and the calculated vehicle’s route are of special interest and are used to select a suitable message carrier vehicle. The algorithm is decentral, each carrier vehicle broadcasts its geographic destination. Then the direct neighbors use information from the navigation system to calculate if and when the geographic position is reached earliest. This value is returned to the carrier vehicle, which in turn forwards a message to the neighbor with the smallest value. If there is no smaller value the message is not forwarded but kept on the current carrier vehicle.

The evaluation is performed on synthetic traces with 2000 vehicles on averaged. There is also an evaluation of the case that only 10% of all vehicles are using a navigation system. GeOpps is only able to route destination nodes with a known geographic position, usually only from vehicles to road side units (’Infostations’), but not in the return direction. A sequel paper [47] deals with the return direction, with the goal to extend the ’range’ of existing WLAN infrastructure. As before, a navigation system which knows the vehicle’s destination is required. Requests to the Infostation are enriched with the calculated vehicle’s route, in order to send the reply message to a geographic position along the vehicles route. This may be by an Infostation (which is rare because of the small number of Infostations) or by C2C routing from an Infostation onto the vehicle’s route. It is important to select a suitable sequence of vehicles based on their predicted routes, in order to ensure that the reply message moves on the route of the target vehicle. To minimize latency the message should be routed to a position that is just a small distance in front of the calculated vehicles position. For the same reason the carrier vehicle should be ahead of the target vehicle and move towards the target vehicle. In the paper very comprehensive examples of the routing process are given, but there is no evaluation.

2.3 Mobility Models and Traces

A simulation based evaluation of routing protocols for mobile networks requires scenarios with (more or less) realistic node movements. There are two basic approaches to generate node mobility. First, synthetic models that use mathematical or algorithmic functions to create mobility traces. These have the advantage of a potentially unlimited duration and number of nodes. On the other hand these models are based on simplified assumptions and therefore unrealistic. The second approach is to record the mobility of real nodes in a real world scenario. Undoubtedly, these so called “traces” are most realistic, but also require a very high effort to record. Therefore, the amount and size of available traces is limited. Traces are usually preprocessed for the use in simulation tools, and for this reason also termed “model”. However, there are also “generative models”, these hybrid approaches use traces to derive models that generate mobility which is statistically similar to the real world mobility from a trace.

2.3.1 Synthetic mobility models

In synthetic mobility models node movements are artificially generated based on random functions and rules that are modeled based on assumptions of 'realistic' behavior. The random walk model and its extension levy walk [48] are widely used. Although the modeled Brownian motion [49] can be observed in various experiments of physics, it is often criticized that the sudden changes of speed and direction are not a realistic behavior of human mobility. Random waypoint [50] introduces a pause before these sudden changes. Similarly, in random direction [51] nodes pause whenever they hit the border of the simulation area, before rebounding from the border at a random angle. However, this leads to a higher node density at the borders. Consequently, the boundless simulation area mobility model [52] maps the rectangular simulation area in a torus. Thereby, nodes disappear at the border and reenter at the same time with the same speed and direction on the opposite end of the simulation area. Another approach to prevent sudden changes of speed and direction is implemented in the Gauss-Markov model [53], which uses a feed-forward of the current node velocity, direction and location for the next timestep.

Several models have been proposed for interacting nodes. For example the Nomadic Community model, in which nodes move from one point to another, the Pursue model, in which nodes move towards a target, and the Column model, in which nodes mimic a search operation.[54] The Reference Point Group model [55] also depicts a search operation (e.g. an avalanche rescue mission), but is more detailed, since it models individual node movement within the group as well as the movement of the group itself.

Map based approaches such the City Section [56] and the Map Based mobility model [57] use real cartographic data to achieve a higher degree of realism in vehicular scenarios. Interaction between nodes is very complex in traffic scenarios, for this reason traffic simulators like SUMO [58] with its microscopic mobility model [59], or MMTS [60] are often used together with real cartographic data to recreate real scenarios for simulation. There are also models that take daily routines into account, e.g. the time variant [61] and the Community model [62]. The Working Day model [57] goes even a step further and depicts social relationships between nodes. It comprises several submodels for home, office and evening activities. Moreover, a transport submodel for walks, cars and buses is included. However, the bus submodel is simplified and was not compared to real bus mobility.

2.3.2 Mobility Traces

Several projects dealt with the generation of mobility models from GPS traces. In [63] position data of taxis was used to model a scenario with about 1000 vehicles in the inner loop of Shanghai. The original data was augmented by interpolation because GPS positions are only updated about every 40 seconds. Moreover, a route-determination algorithm was applied to fill the gaps between position updates. However, the characteristics of taxis are not comparable to buses. In [64], Zhang et al. used traces of a bus-based DTN-testbed with 40 vehicles to derive a generative model of inter-contact times. Traces from the same testbed were also used in [39]. VANET simulations based on bus movement traces with a significantly larger number of vehicles were performed in [65]. The same trace was used in [66] to evaluate a cluster-based DTN forwarding algorithm. This trace was collected in 2001 in Seattle and includes over 1200 vehicles. To the best of my knowledge there were no other publicly available large scale traces for the simulation of bus-based DTNs available besides [67]. Therefore I recorded an additional trace [68] from continuous real-time data publicly available via an API from the Chicago Transport Authority. The data contains 1971 buses, 150 routes and 11701 bus stops. This trace is presented in detail in chapter 3.2 on page 51.

2.4 Summary

DTN is a promising concept for communications in public transport. In this chapter several use-cases and application scenarios were introduced. Besides applications that are related to the operation of public transport, e.g. passenger information, there are also applications that can be provided as a service, e.g. environmental monitoring. There are several approaches to DTN routing, ranging from keep-it-simple concepts (e.g. flooding) to very sophisticated probabilistic algorithms (e.g. Prophet). However, these existing routing algorithms follow a generic approach, so that the routing algorithm is applicable to as many diverse scenarios and applications as possible. But as already discussed in chapter 1, several different routing algorithms can coexist in a DTN node. Therefore, implementing a specialized routing algorithm which is optimized for the characteristics of public transport is a viable and promising approach. However, such a routing algorithm has not been proposed before, which may also be caused by the lack of suitable mobility traces.

Figure 2.14: Solar-powered DTN node.

Chapter 3

Communication Characteristics and Mobility Analysis

In this chapter the background of communication in public transport DTNs is analyzed. For the development of routing mechanisms it is important to understand what exactly happens during communication in a real world scenario, and if the envisioned applications are feasible at all. Besides gaining a deeper understanding of such a system's requirements the following analysis also serves the purpose of obtaining sustainable values for the parameters (e.g. communication range and channel capacity) of the later simulation based evaluation

3.1 Communication Characteristics

There are several wireless technologies that can be used for inter-vehicle communication. For vehicular DTNs in public transport IEEE 802.11a/b are the most promising candidates of the various WLAN sub-specifications, because these technologies can be deployed in a very economic way. For example they do not require licensing and the hardware is widely available and therefore relatively cheap. Many public transport vehicles already have the hardware for these technologies installed [1]. However, these systems are currently only used for stationary and semi-manual processes. Trip information, for example is transmitted to a vehicle's on-board passenger information system via WLAN at some transport operators. This currently happens when a vehicle leaves the garage at the beginning of shift, by actively stopping near an access point. The driver manually triggers the update process by pressing a button and then waits for the data transmission to finish. This real use-case already shows that a DTN based system would be beneficial, but also that IEEE 802.11a/b has certain limitations.

In the following the suitability of IEEE 802.11a/b for dynamic vehicular communication is examined. The goal is to check the basic feasibility and to gain realistic parameters for the system design and the later simulation-based performance evaluation. Range and data rate are of special interest, but also the general characteristics such as link stability and signal-to-noise ratio. Since all these parameters are influenced by the very complex signal propagation in mobile urban environments we chose an experimental evaluation instead of a more-or-less unrealistic simulation. [17, 69]

3.1.1 IEEE 802.11b real-world Performance Evaluation

For evaluating WLAN based vehicular DTNs' applicability and performance in real-world scenarios we implemented a prototype system based on off-the-shelf notebooks. These prototypes are only intended to demonstrate the principles of future operation. The notebooks are equipped with an 802.11b WLAN PC Card, a low cost omni-directional WLAN antenna with 3m cable extension and an USB GPS receiver supporting DGPS for higher accuracy. To obtain environmental data for this demonstration we used a temperature sensor attached to the serial port of the notebook. The prototypes have been installed in vehicles, with the external antenna being placed on the front edge of the vehicle's roof.

The road test comprises three scenarios to asses the main aspects of the communication performance in vehicular environments: the connectivity while two vehicles pass each other (A), the connectivity in urban scenarios (B) as well as the message delivery in vehicular DTNs (C). The setup of these three scenarios is described in the following.

In scenario A, we analyzed the behavior of 802.11b WLAN with two vehicles driving past each other for determining the maximum communication range and the communication characteristics. For these measurements, we chose a straight, secluded road on the outskirts of the city of Braunschweig. The road section has a length of 1200m and is lined with trees along one side.

The road was not used by other vehicles during our measurements. It is mostly planar with only a small depressed area near the middle of the driving distance. All in all, the environment made it possible to obtain comparable results of different measurement runs, since dynamic influences of e.g. other vehicles or obstacles are minimized. The same setup was used in scenario B, for analyzing the performance of message delivery between two vehicles. Finally, in scenario C we also investigated the performance of 802.11b WLAN with two vehicles exchanging messages in a residential area in Braunschweig. This area is characterized by a grid-like road topology. It is covered with multi-story apartment houses that prevent a direct line of sight between two parallel road sections. Moreover, cars were parked along the street, pedestrians walked on the sidewalk and vehicles drove through the streets. Due to the multitude of varying parameters, these tests are not reproducible and can only give an impression of the qualification of 802.11b WLAN in urban scenarios.

A special tool ("gpsping") was developed to determine the signal strength and data rate of each received packet as well as the vehicle's position at that time. In the following, we present the results of our practical evaluation in detail.

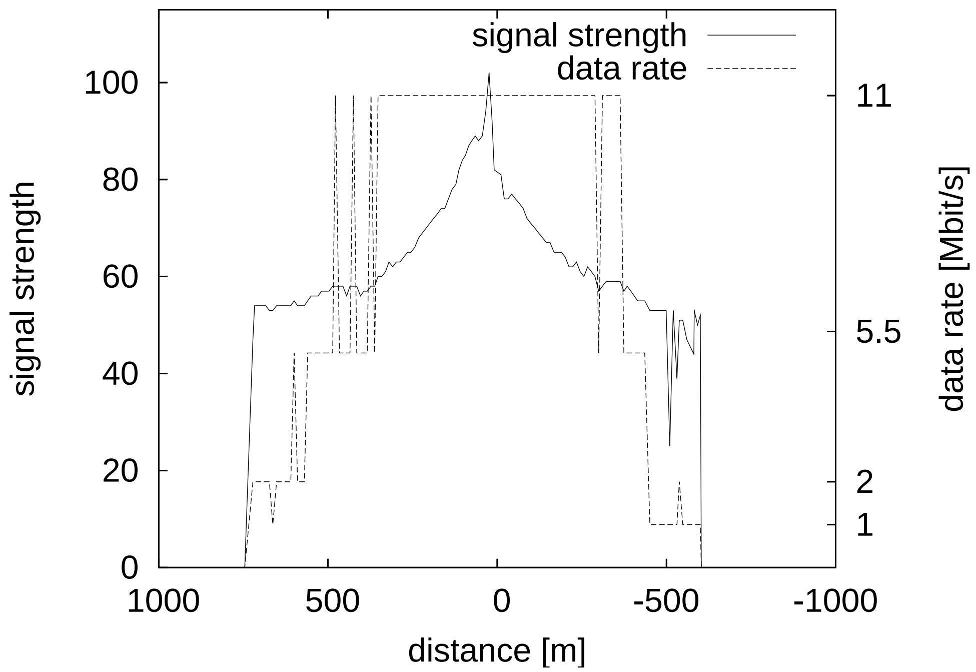

Figure 3.1: 802.11 performance at v=20km/h.

3.1.1.1 Signal Strength and Data Rate

The main focus of scenario A was the behavior of 802.11b WLAN in scenarios with high mobility. Special attention was turned on how long two vehicles driving past each other are able to exchange data. To measure the signal strength and the WLAN data rate we equipped two cars with laptops, WLAN cards, external antennas and GPS receivers. Each of these laptops ran an instance of “gpsping”. The signal strength is given by the unitless signal strength indicator reported by WLAN card’s driver.

Figures 3.1-3.3 show measurements with two vehicles driving past each other at a speed of 20km/h, 50km/h and 80km/h. The graphs are normalized to the distance between the vehicles for comparing the characteristics of signal strength and data rate at different vehicle speeds. At the beginning of a measurement run, the vehicles are positioned at opposite ends of the road, at the maximum distance of 1200m. During the measurement, the vehicles pass each other. This point corresponds to a distance of almost zero in the graph. A negative distance value means that the vehicles have already passed each other. The distance is computed on the basis of the vehicles’ position and the high precision time signal, which are both obtained from the GPS receiver. The GPS position update rate is 1Hz, and for this reason the resolution of the measurements depends on the vehicle speed. Therefore the 20km/h measurement run has four times the resolution of the 80km/h measurement run. As a side effect the WLAN data rate appears to fluctuate more heavily at 20km/h.

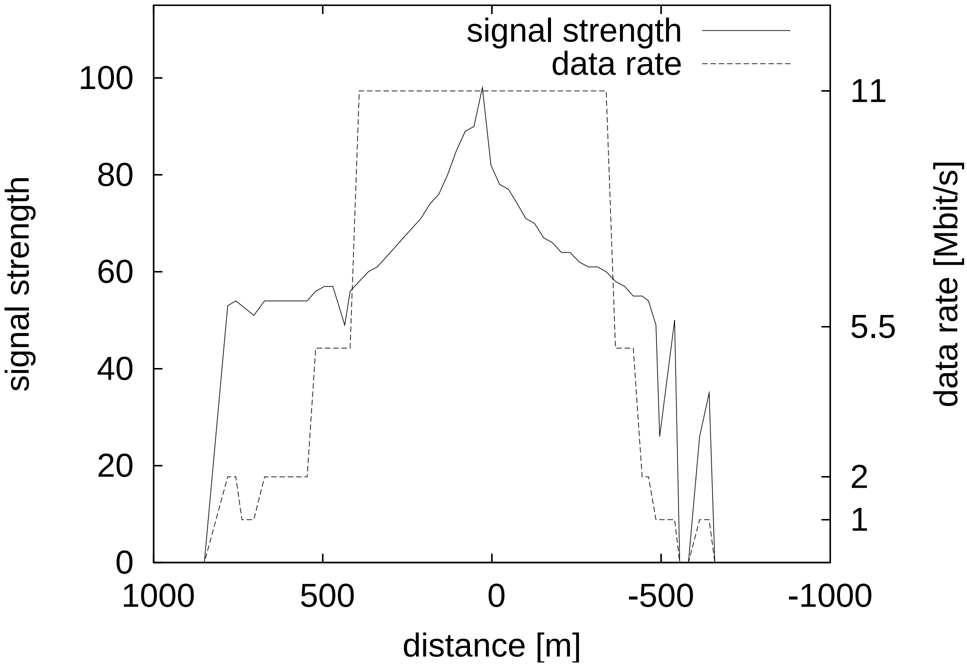

Figure 3.2: 802.11 performance at v=50km/h.

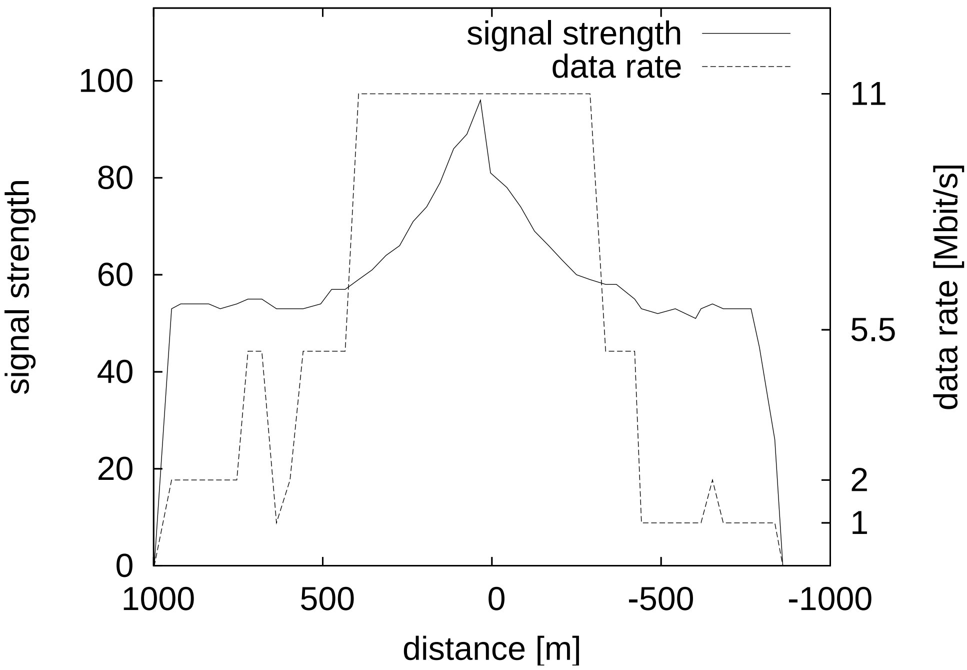

Figure 3.3: 802.11 performance at v=80km/h.

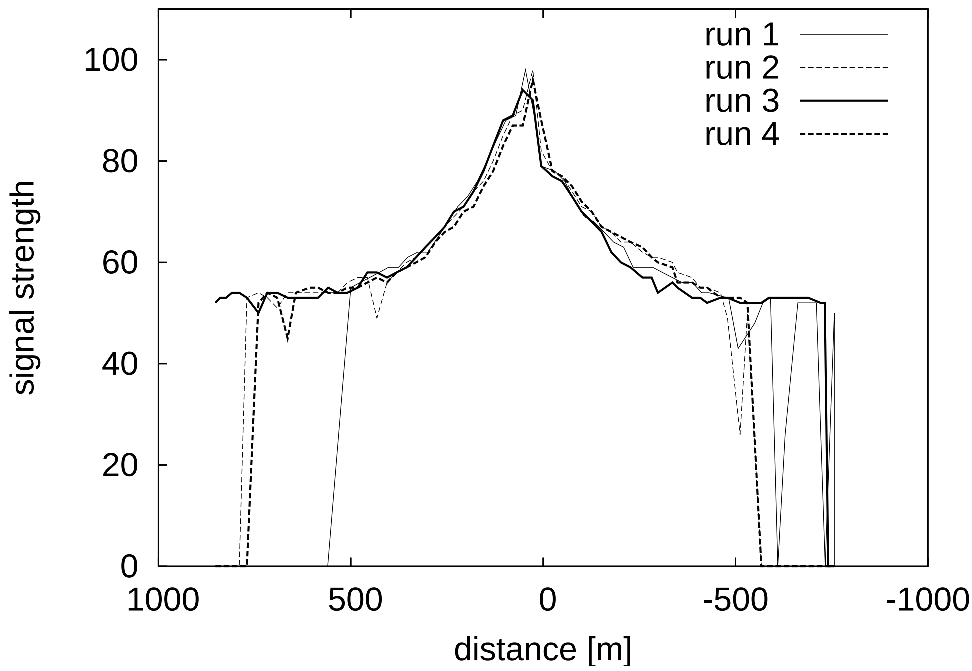

Figure 3.4: Comparison of measurements at the same vehicle speed: signal strength.

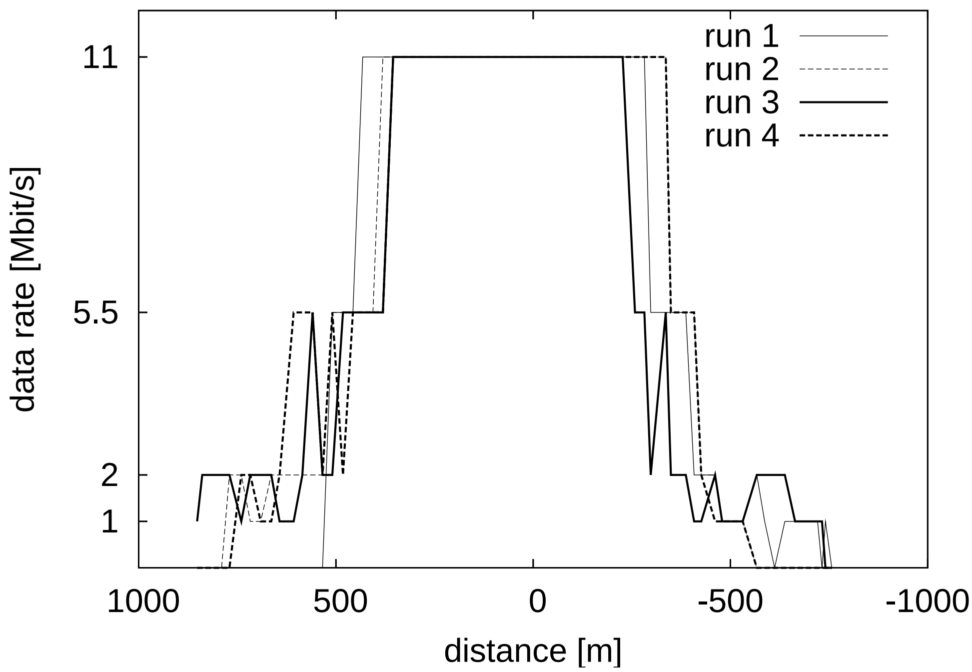

Figure 3.5: Comparison of measurements at the same vehicle speed: data rate.

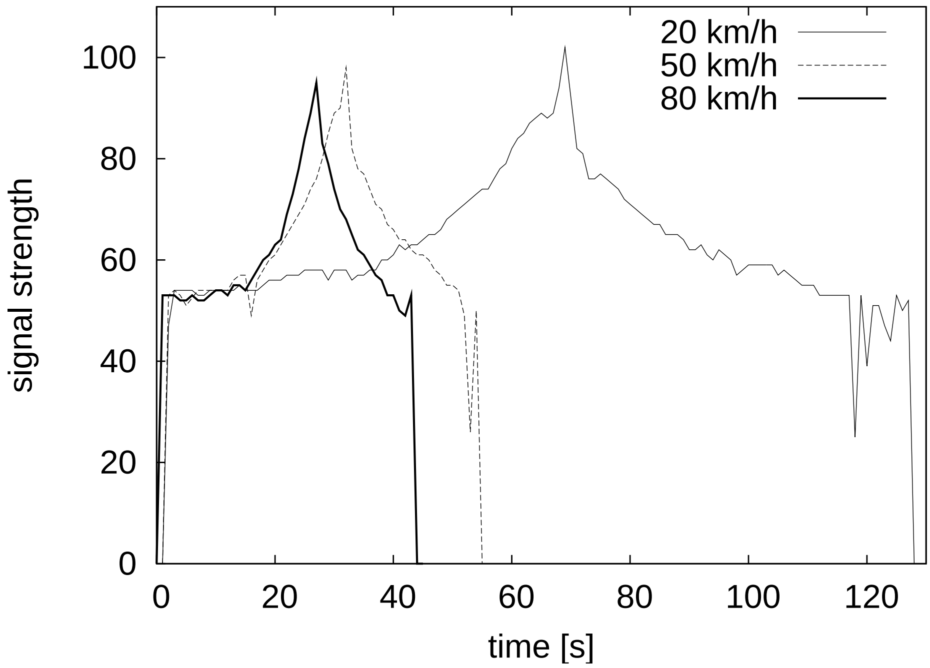

Figure 3.6: Comparison of signal strengths at different speeds: time.

In figure 3.1 it can be observed that the range is well above 500m. Both range and the signal strength are slightly asymmetric because the antennas are mounted at the front side of the vehicle's roof, which causes a shadowing effect on its rear side. The signal strength graphs in figures 3.1-3.3 show a characteristic uniform peak when both vehicles meet. During all measurement runs at a specific speed the maximum WLAN data rate is available for a continuous period of time, with durations of 61s at 20km/h, 28s at 50km/h and 17s at 80km/h. For each run there is also a period of time in which the link operates at a lower data rate but is still available for transmissions. This period is roughly as long as the period of full data rate.

Figures 3.6-3.7 illustrate the signal strength at different speeds. Figure 3.6 gives a good impression of the durations of the measurements. The signal strength graph in figure 3.7 is normalized to distance and shows how similar the peaks are even at different speeds. In figure 3.4-3.5 it can be observed that the results of different measurement runs at the same speed are reproducible.

The probability distribution of the WLAN data rate in dependence of the distance is shown in figure 3.8. It is calculated on the basis of all measurements. For distances of <333m the probability for a data rate of 11Mbps is 92%. It decreases to 57% for distances of 666m respectively 47% for distances below 1000m. It should be mentioned that distances below 202m always resulted in 11Mbps links in all measurement runs.

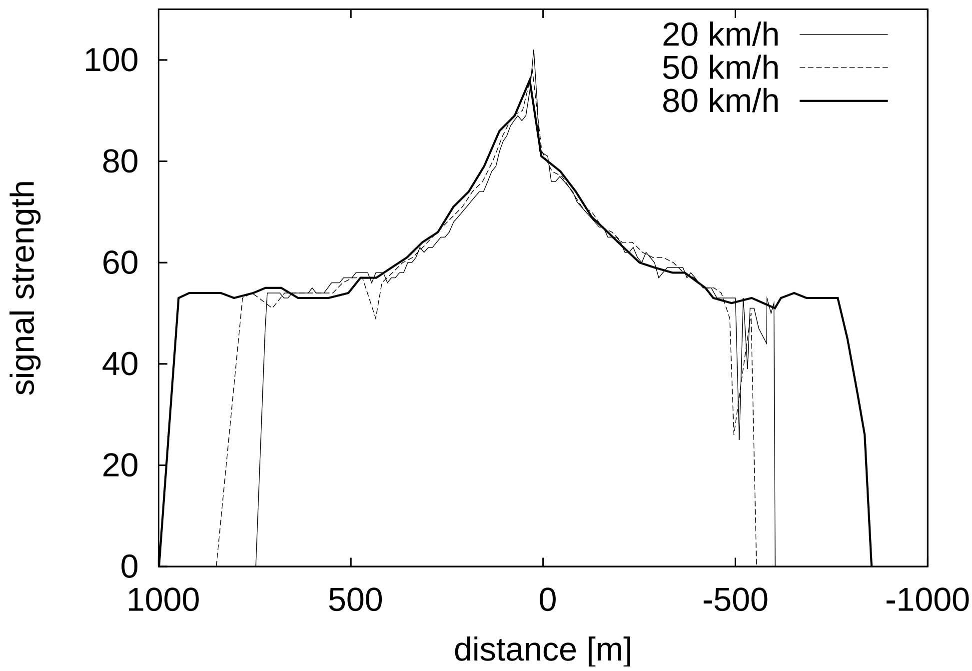

Figure 3.7: Comparison of signal strengths at different speeds: distance.

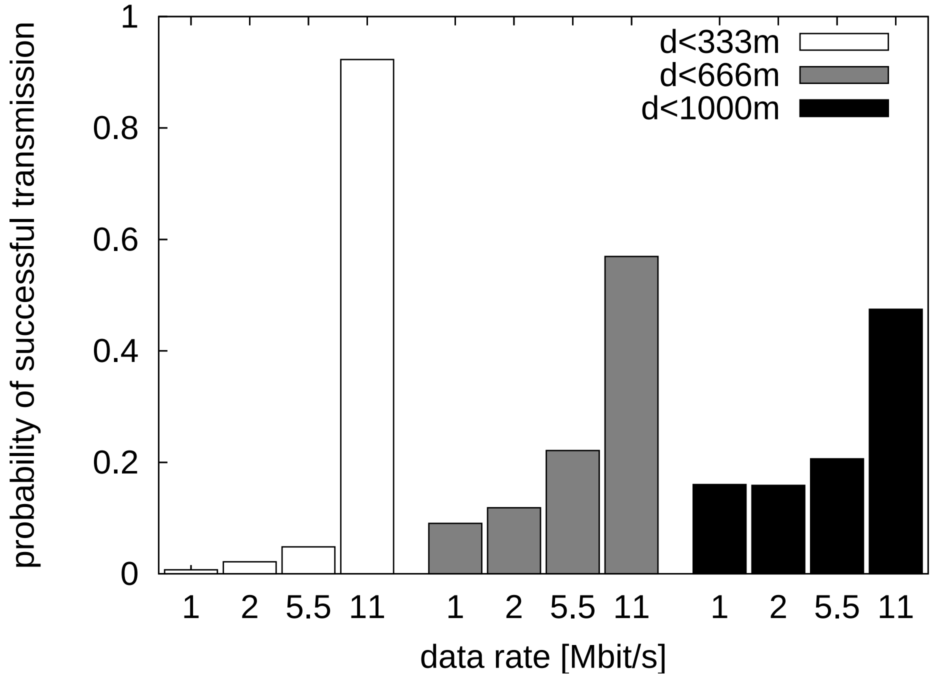

Figure 3.8: Probability distribution of 802.11 data rate.

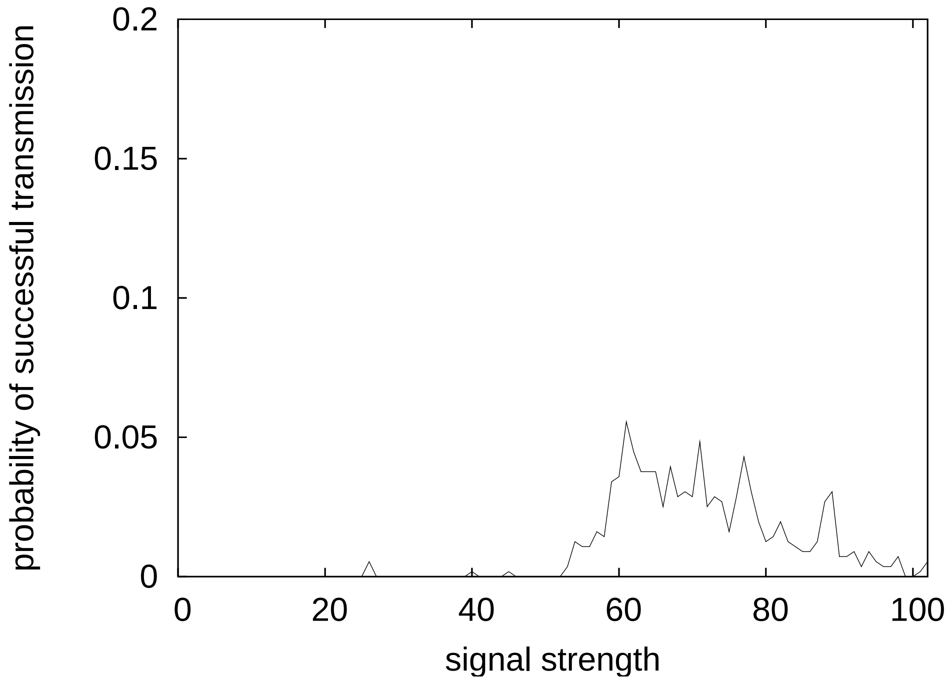

Figure 3.9: Probability distribution of the signal strength: d≤333m.

The probability distribution of the signal strength is depicted in figures 3.9 and 3.10. Figure 3.9 shows the distribution of the signal strength for distances lower than 333m. It can clearly be seen that the measured signal rate at this distance is mainly higher than 60 which means that the connection between the vehicles is very stable and transmissions can take place at high data rates. Hardly any lower values have been measured in our scenario, which results from packets that were lost due to transmission errors during the measurement. Figure 3.10 shows the cumulative probability distribution for the whole drive-by scenario. The graph illustrates that while both vehicles pass each other, most of the signal strength values are in an interval of 50 and 65. At signal strengths lower than 24 no connection is possible.

3.1.1.2 802.11b Range in Urban Scenarios

A second field of interest is the transmission range of 802.11b WLAN in urban environments, since these are typical for public transport scenarios. The results of this scenario are of major importance for the usability of WLAN hardware in vehicular DTNs. To gather some real life data, two cars were equipped with the same hardware components described above and drove through a densely settled residential area in Braunschweig.

Figures 3.11-3.14 illustrate exemplary traces of the two cars while driving through the city. The arrows point in the driving direction while the dots indicate that a WLAN link is available between the two nodes. The link status was probed once a second.

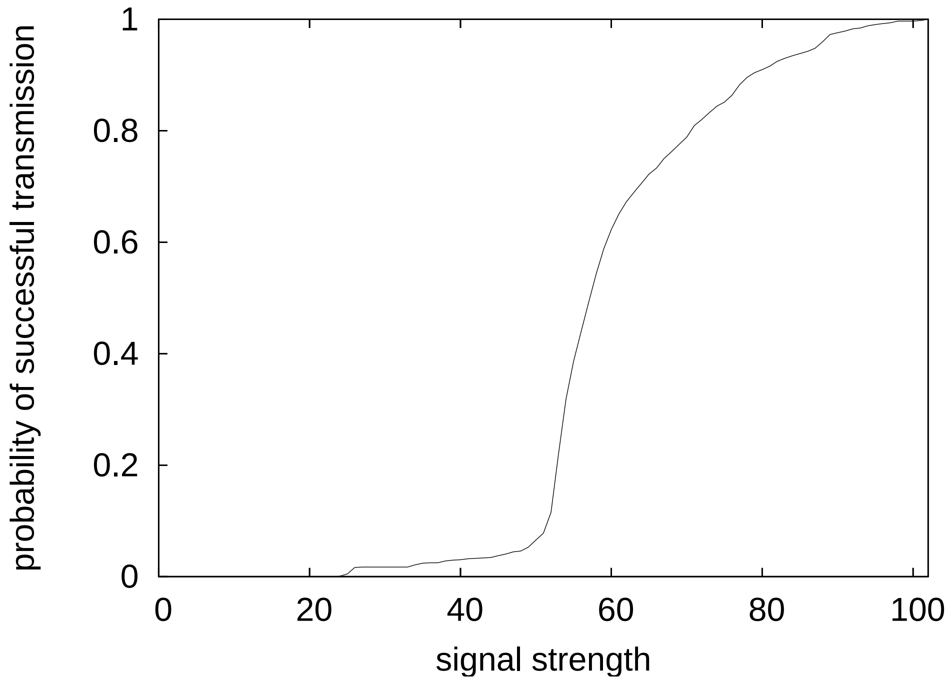

Figure 3.10: Probability distribution of the signal strength: cumulative probability d<1000m.

At the beginning, both vehicles start at the same place and head towards the first intersection where the vehicles' routes split up (figure 3.11). After one vehicle turned right at the first intersection, the direct line-of-sight was lost. However, the WLAN link remains available for eight seconds. The transmission range is about 100m at this intersection. The same effect can be observed at the next intersection. A connection can be established about seven seconds and 118m before both vehicles meet. While passing this intersection, the vehicles were able to exchange data during a period of 16s (figure 3.12). Thereby, a data rate of 11Mbit/s was available for 13s even without having a direct line-of-sight during the whole period of time. At the next intersection the vehicles can communicate over a distance of 231m along the same road section. Figure 3.13 illustrates that the vehicles stay connected for about 30s (with short disruptions) although the direct line-of-sight is interrupted by a row of multi-story buildings. A further connection between the vehicles becomes available for about 5s when both pass the next intersection at the same time. The intermittent connectivity after both vehicles passed the intersection can be explained by multipath propagation effects as well as vacant lots. Finally, figure 3.14 gives an impression on the WLAN transmission range along straight roads. The connection is established even before the vehicles arrive at the intersection and lasts for about 40s. The maximum transmission range was 171m.

In summary it can be stated that the results in the urban scenario clearly differ from the results obtained on the secluded road outside the city with respect to the transmission range and thus to the duration of the connection. However, this is not unexpected since the radio propagation in built-up city areas is highly affected by dynamic changes in the environment like other road users. Nevertheless, the results show that WLAN is suitable for deploying DTN approaches like EMMA even in urban scenarios. The vehicles were able to exchange information at the maximum data rate of 11Mbit/s during almost the whole connection period. This is due to the fact that the signal strength increases faster than in the scenario on the secluded street when both vehicles get into transmission range. The signal attenuation mainly results from multipath propagation effects and not from the physical distance of the vehicles. In this scenario, vehicles were able to exchange data over distances of more than 100m, even without direct line-of-sight. Of course, we do not expect these results to be representative for urban areas in general. However, our measurements have been performed in a very challenging environment. Since bus or tram routes often follow main roads, we can assume that in these scenarios the outcome may be even more encouraging. All in all, the results are promising for realizing EMMA in urban areas.

3.1.1.3 Feasibility of 802.11b based DTN measurements

In the previous scenarios, we focused on the performance of 802.11b WLAN in urban vehicular scenarios. Although these results are very encouraging, they do not prove the suitability of DTN for vehicular applications in general and the DTN reference implementation[1] in particular to set up a distributed measurement network. To investigate this in more detail we equipped two measurement vehicles with the adequate software and performed tests on the same secluded road. Two vehicles were passing by each other while exchanging bundles.

Figures 3.16 and 3.17 show some exemplary results of this measurements for vehicle speeds of 20km/h and 50km/h respectively. The number of exchanged bundles is aggregated in 10s intervals. As expected, the number of exchanged bundles increases with the signal strength. At a speed of 20km/h it was possible to exchange 3545 bundles during a period of 150s. At 50km/h the connection lasted for 60s which was enough time to transfer 1943 bundles. The resulting average data rates of 1087Byte/s at 20km/h and 1455Bytes/s at 50km/h are clearly lower than it could be expected on the basis of the previous measurements, possibly due to the development stadium of the DTN reference implementation. We observed that the reference implementation handles the bundle exchange in a rather inefficient way. It sends a burst of bundles and then starts to look for other potential communication partners within range. For small bundles this behavior wastes a lot of performance since links in vehicular environments persist only for a short period of time.

To explore the usability of DTN for a distributed measurement network we set up four test vehicles each equipped with a WLAN enabled notebook running the DTN software. One node acted as the central evaluation server (CES) being placed at one side of our test site. The other three nodes obtained sample data. One of them acted as a stationary node at the far end of the test site outside the CES' range. The remaining two acted as

[1] http://sourceforge.net/projects/dtn/

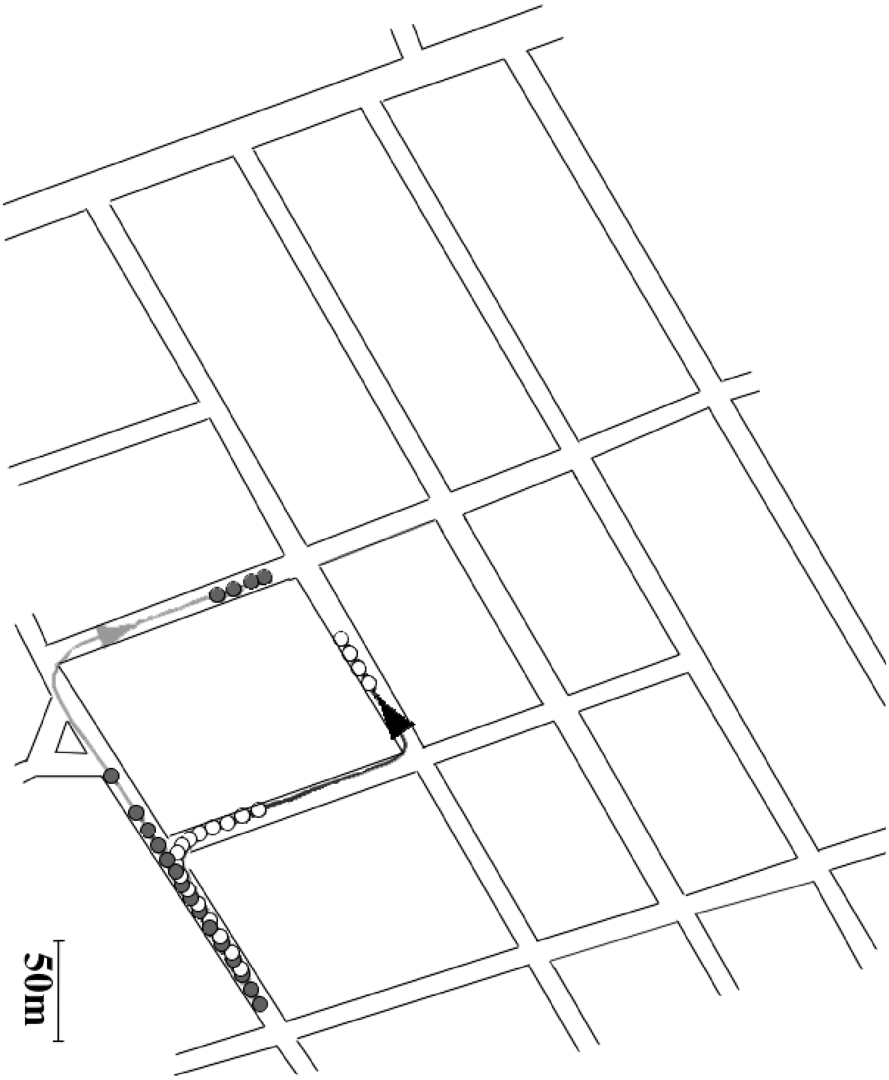

Figure 3.11: WLAN connectivity in urban scenarios: both vehicles start at the same place and head towards the first intersection where the vehicles' routes split up. Non-line-of-sight communication is observable at the intersections. Vehicle speed is typical for urban traffic (up to roughly 50km/h).

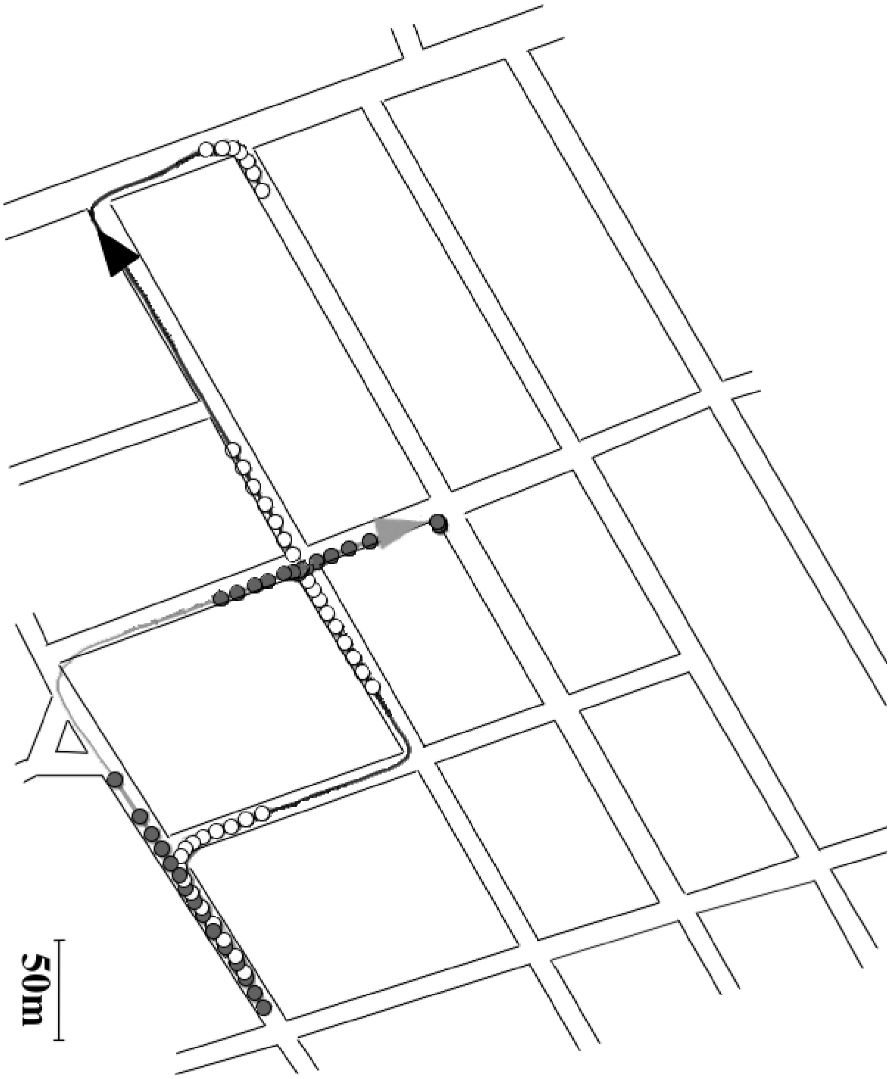

Figure 3.12: WLAN connectivity in urban scenarios: A non-line-of-sight connection is established about seven seconds and 118m before both vehicles meet at the next intersection. While passing this intersection, the vehicles were able to exchange data during a period of 16s.

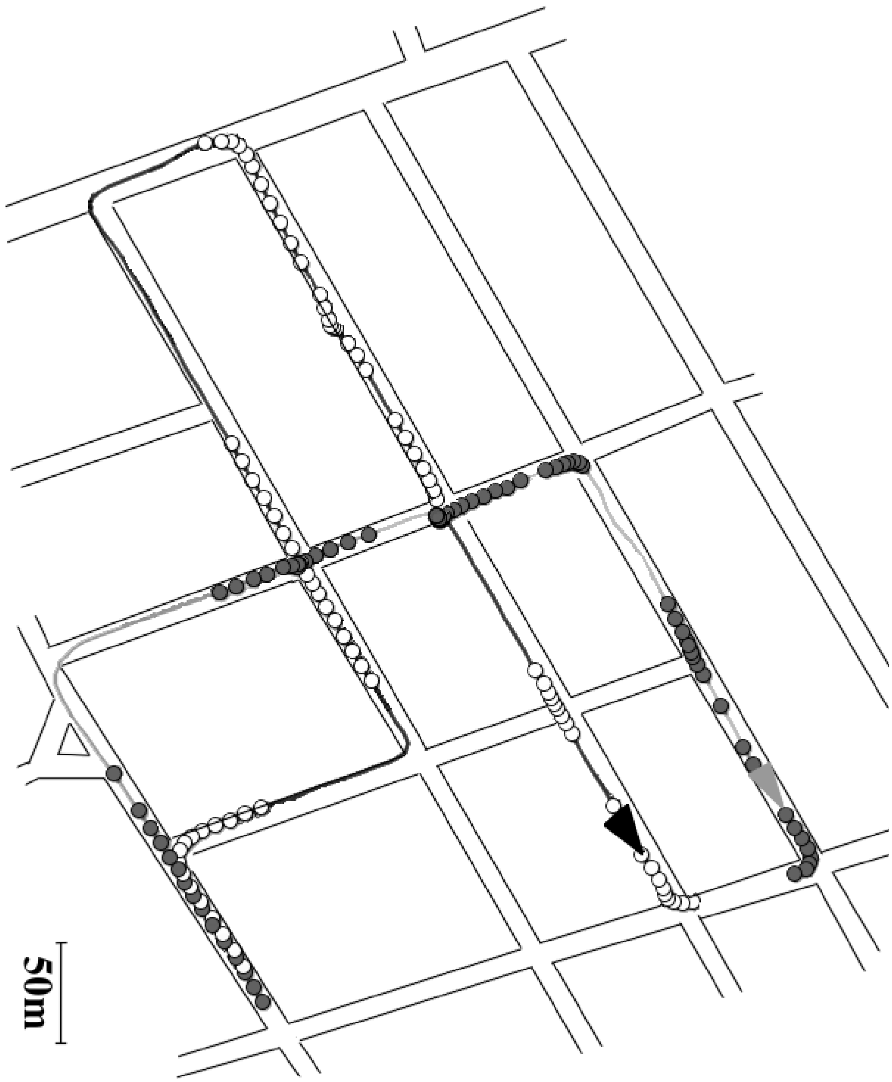

Figure 3.13: WLAN connectivity in urban scenarios: Although there are disruptions, the vehicles stay connected for roughly 30s, even while the direct line-of-sight is interrupted by a row of multi-story buildings.

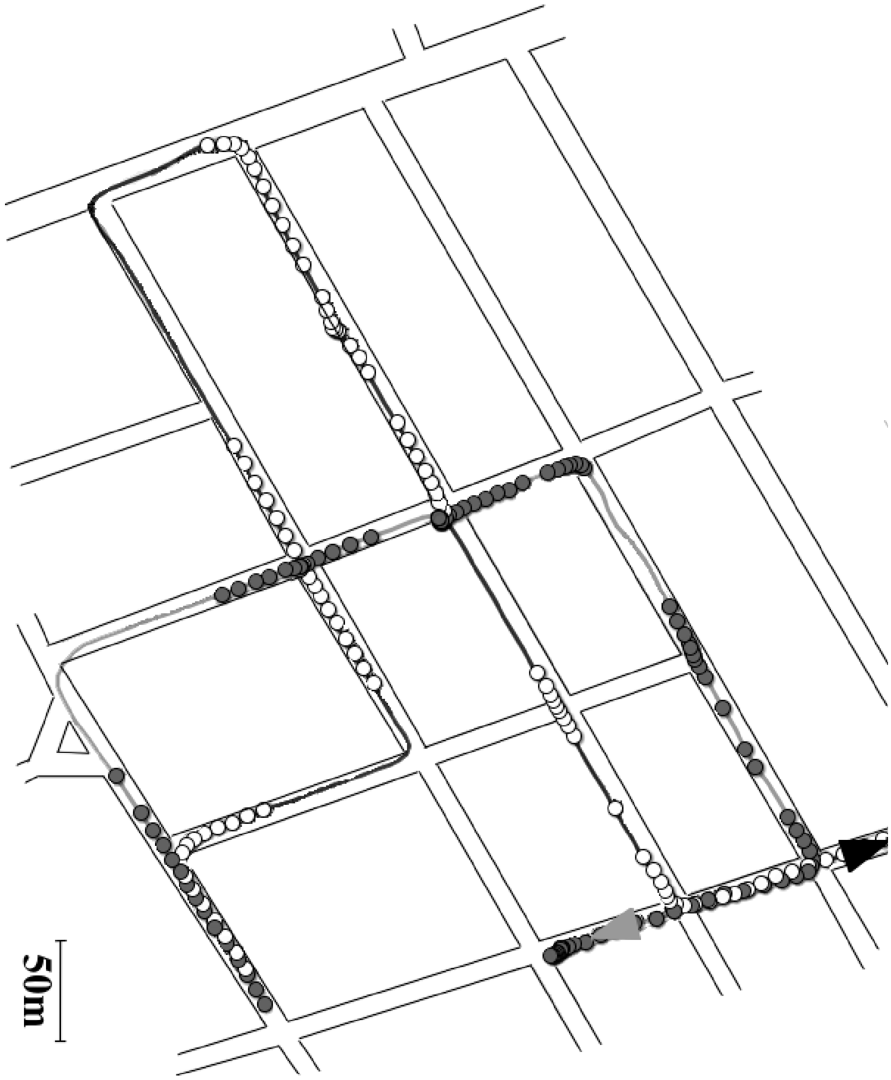

Figure 3.14: WLAN connectivity in urban scenarios: Transmission range along straight roads. The connection lasts for about 40s with a maximum transmission range of 171m.

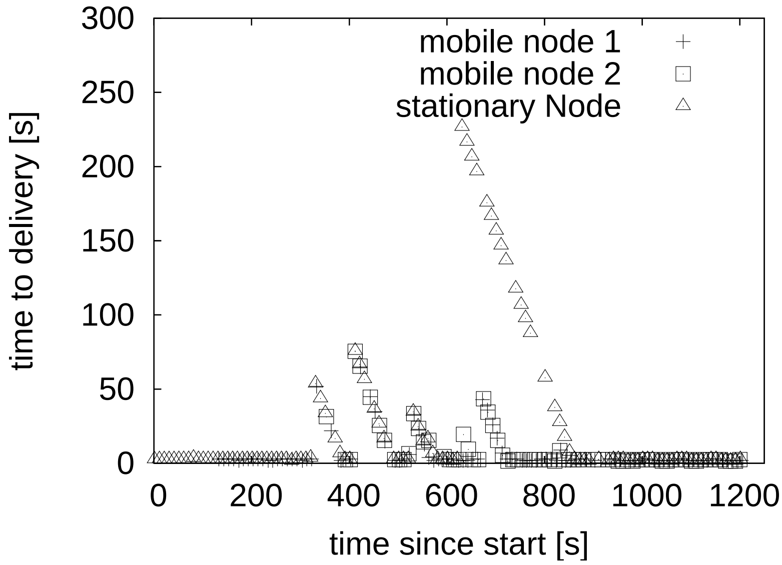

Figure 3.15: Delay of bundles created by different nodes.

mobile nodes driving predefined routes on the test site with a size of 400m×500m. In figure 3.15 you can see the time it takes to transfer a new bundle to the CES. The average delay of bundles created by the stationary node is 26.28s, the minimum is 1.77s and the maximum is 226.79s. The fairly low minimum is caused by a mobile node directly relaying between the stationary node and the CES. The average delay of bundles created by the two mobile nodes is 7.44s, the minimum is 1.47s and the maximum is 75.45s. This evaluation proves the basic feasibility of vehicular DTNs with IEEE 802.11b hardware.

3.1.2 IEEE 802.11a in Public Transport Scenarios

In the previous paragraphs IEEE 802.11b was evaluated. Another commonly used license exempt standard is 802.11a. Both standards share the same MAC-layer, but define different PHY-layers. 802.11b uses the 2.4GHz band with Direct Sequence Spread Spectrum (DSS) modulation, while 802.11a operates around 5.5GHz with Orthogonal Frequency Division Multiplexing (OFDM). There are several successor standards and proprietary extension that specify various data rates, but a detailed discussion is out of scope of this work. The main difference that is of interest here is the different radio frequencies. From a radio propagation point of view there should be a clear advantage of 802.11b because of the lower frequency that leads to lesser attenuation. However, from a practical point of view, there are two major advantages of using 802.11a. First, for regulatory[2] reasons a significantly higher transmit power is allowed for 802.11a in outdoor deployments. And second, because most consumer products operate in the 2.4GHz ISM band, there is a lot of interference. Therefore, 802.11a is an attractive

[2] This is valid at least for the German / European regulations. There are frequency allocation standards by ETSI, FCC and various country specific legislation that may differ.

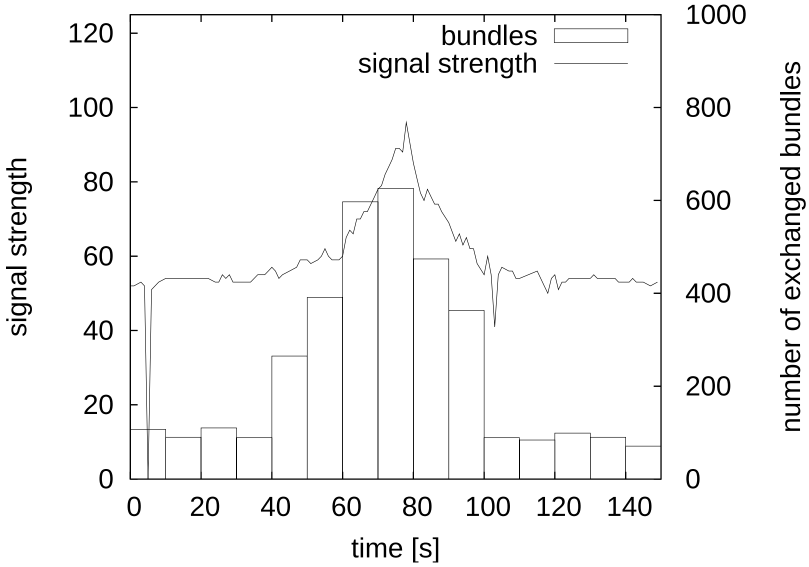

Figure 3.16: DTN bundle exchange at different speeds: v=20km/h.

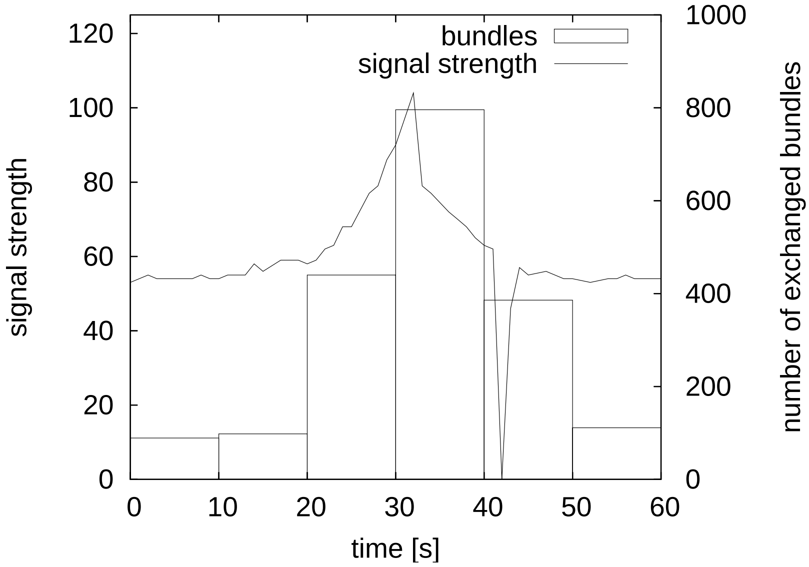

Figure 3.17: DTN bundle exchange at different speeds: v=50km/h.

candidate for performance reasons. Moreover, 802.11a operates near the band of the upcoming 802.11p standard for vehicular communications. For this reasons we also evaluated the suitability of 802.11a for vehicular DTNs [69].

Vehicular public transport DTNs will likely use the omnipresent IEEE 802.11 technology to communicate between vehicles. IEEE 802.11b/g has been state of the art for home networks in the 2.4GHz band and now the 5GHz band is used as well. Especially for vehicular networks, IEEE 802.11p is in development. Unfortunately, so far there are neither off-the-shelf hardware systems nor thorough evaluation studies of such 802.11p systems. Since 802.11a is the closest relative, we use its performance characteristics as basis for simulation based evaluation. To the best of our knowledge, no measurements of 802.11a in a public transport environment have been published yet. Therefore, we have conducted measurements to determine the necessary parameters for simulations of DTNs.

To obtain similar information for IEEE 802.11a we have used the following setup. We placed a static node 3m above the ground and installed a mobile node on the roof of a vehicle. The nodes have used antennas with a total path gain of 11dBi. Both stations used a Ubiquiti XR5 [70] IEEE 802.11a WLAN card. In order to measure the bandwidth of the WLAN connection on the application layer, we have used iPerf [71] with TCP.

To test the communication characteristics, the vehicle was placed 350m away from the station. It starts to drive towards the station, uses a roundabout to turn around in front of the station and then drives away again. The targeted speed is 45km/h which is a usual speed for inner city vehicles in Europe. The distance of 350m was chosen since it was the maximum distance over which communication was possible. The measurements were performed at a street in an industry park with trailers and buildings located right next to the street and can therefore be characterized as a semi-controlled environment. The sender and receiver were always in direct line of sight and no other traffic was present.

Figure 3.18 shows the distance between the sender and the receiver. The w-shape is caused by the roundabout next to the static node. In addition, the figure shows the bandwidth that could be achieved on application level in one of the measurements. The fluctuations are due to interference and multi-path propagation caused by surrounding objects. The average bandwidth of all measurements is 1587.88 Kbytes/s.

3.1.3 Discussion of the Results

The field tests were set up to determine whether 802.11a/b WLAN is suited for vehicular DTNs. The transmission range and the data rate were evaluated on a secluded road. Our results show that a transmission range of more than 500m is feasible using standard hardware components. The measured value is based on the assumption that a direct line-of-sight is available. In the urban scenario, we found out that communication is still possible over a distance of more than 100m even if no direct line-of-sight is available. The comparison of various result sets has shown that the measured values were reliably reproducible. Finally, we analyzed the feasibility of DTN-based message delivery for our application.

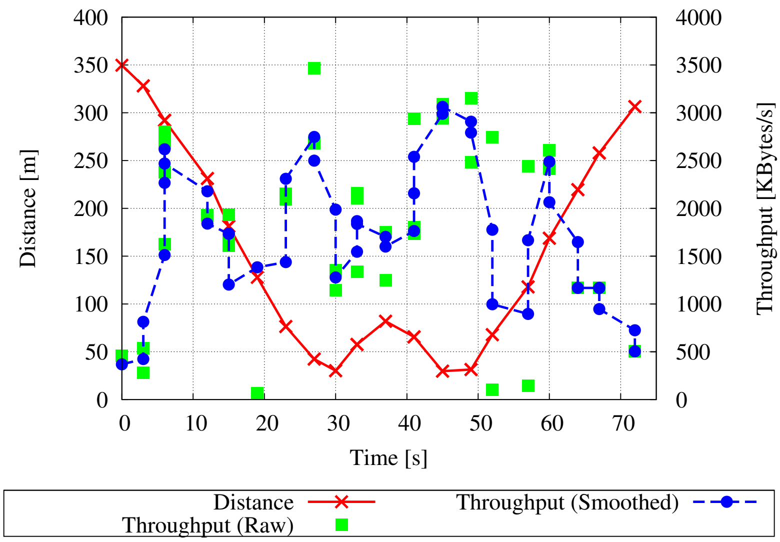

Figure 3.18: Sender-Receiver distance and achievable application layer throughput over measurement time for one exemplary measurement.

The evaluation results are very encouraging, since they clearly show that the net data rate of 802.11b WLAN is high enough to exchange a significant amount of data (several tens of megabytes) while driving past each other at the speed of common overground public transportation vehicles. While the performance of the experimental DTN implementation does not yet suffice to operate at full WLAN capacity, we have seen that the system already works well in urban environments.

3.2 Mobility Analysis

This work focuses on urban public transport systems with vehicles operating on surface infrastructure, especially buses running on roads. These vehicles move on lines along stops. The arrival and/or departure time at the stops is usually defined by a timetable, which also defines a frequency of service on a given line. However, there are also some transport systems that only define the frequency of service but no binding arrival/departure times. Usually the frequency of service changes several times a day, according to a static plan based on the average number of passengers at a given time. For example frequency of service may be five minutes during rush-hour and 30 minutes at late evening. It should be noted that timetables are hard to maintain since unpredicted disturbances may happen such as traffic congestion, vehicle failure, traffic accidents or blocked roads. Another important property is that there is no fixed assignment of vehicles to lines. Usually each vehicle is dispatched to a random line at the beginning of a shift. [64] Moreover, during one shift vehicles frequently change lines in order to increase usage rate by reducing deadheads and waiting times.

A possible contact occurs whenever two vehicles are in each others radio range. Vehicles move on lines and follow a timetable, therefore potential contacts are roughly predictable. However, there are always minor variations, e.g. caused by traffic lights or changing numbers of passengers getting on and off at stops. In certain situations these minor variations make the contact prediction very difficult. For example a variation of just a few seconds can prevent a potential contact between vehicles of two different lines crossing at an intersection. For this reason it is important to assess the probability and the duration of a possible contact. Since all vehicles move on predetermined lines a prediction based on the length of joint segments can be made. However, vehicles can also move in different directions on the same line. A contact is most likely if both vehicles enter a joint segment simultaneously on both ends and move in opposite directions. But these contacts can also be expected to have a shorter duration than contacts of vehicles moving in the same direction, since vehicle speeds in joined segments are very similar. Therefore, vehicles driving in the same direction with a low distance will maintain longer contacts.

3.2.1 Suitable Models and Traces

This section gives an overview of the models and traces that are available for analysis in order to gain a better understanding of the dynamics of public transport vehicles. Moreover, these models and traces are also needed for realistic simulation-based evaluation. For this purpose a trace has to contain at least a reasonable number of time/position-datasets for each vehicle. Unfortunately, there are not many traces of public transport buses available, and some are lacking important data. [72], for example, was recorded with iMotes mounted to buses. Bluetooth contacts with other iMotes on other buses were recorded. However, the vehicles' locations were not recorded. Therefore, it is not possible to analyze in which situations contacts occurred (e.g. at stops or at intersections) and if there are, e.g., areas in which contacts are very (un)likely. Moreover, without position data it is not possible to determine the influence of vehicle speed, or to examine the

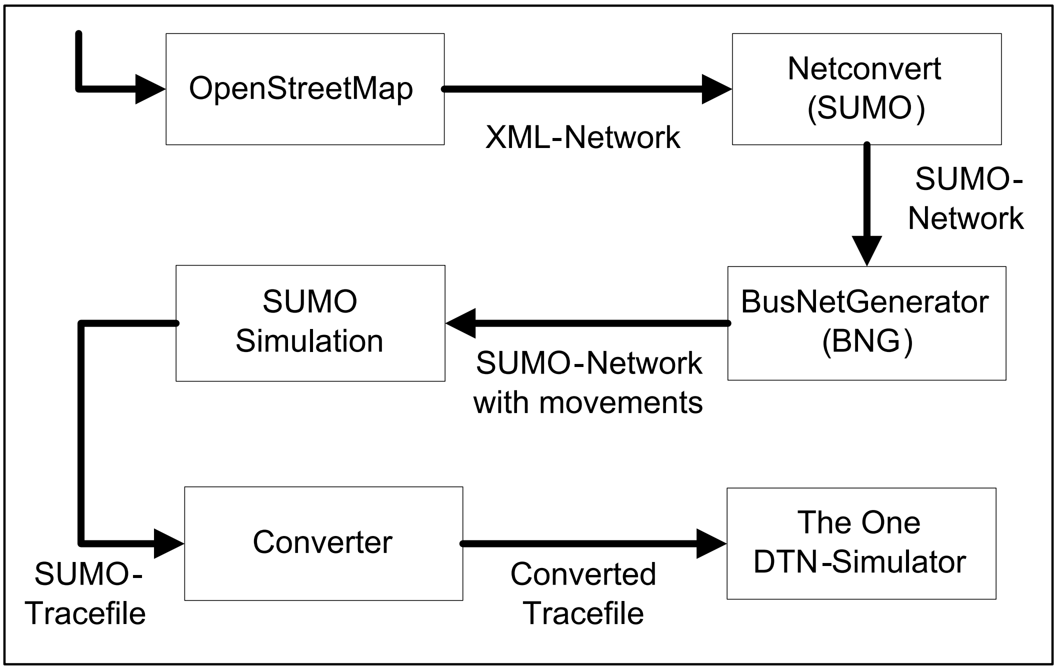

Figure 3.19: Toolchain used to generate the synthetic Braunschweig model

range of the used radio technology.

The available traces that are suitable for the research in this thesis are described in the following sections. First, the Braunschweig Model is introduced, which is based on a micromoblity simulator with real cartographic data and public transport timetables. Second, a real-world trace of buses in Seattle is available from the Crawdad Repository[3] is introduced. And third, we present a real-world trace that contains GPS coordinates and extensive metadata of roughly 2000 buses in Chicago.

3.2.1.1 Braunschweig Model

The Braunschweig Model is a synthetically generated trace based on the SUMO [58] micromobility simulator, cartographic material from the OpenStreetMap [4] project, and data on routes, lines, timetables and stops from Braunschweig's public transport operator[5]. In the following the process of creating a synthetic trace is described.

The toolchain used to create the synthetic mobility model is shown in figure 3.19. First, cartographic material from OpenStreetMap of the area of Braunschweig is exported to SUMO with the tool 'Netconvert'. Second, the generated scenario is supplemented by the corresponding lines, stops, vehicles and their timetables. For this purpose the tool 'BusNetGenerator' [73] was developed. To add the required data in a simple way, it offers a GUI with Google-Maps to place stops in the network. Subsequently, the lines can be

[3] http://crawdad.cs.dartmouth.edu
[4] http://www.openstreetmap.org
[5] http://www.bsvag.de

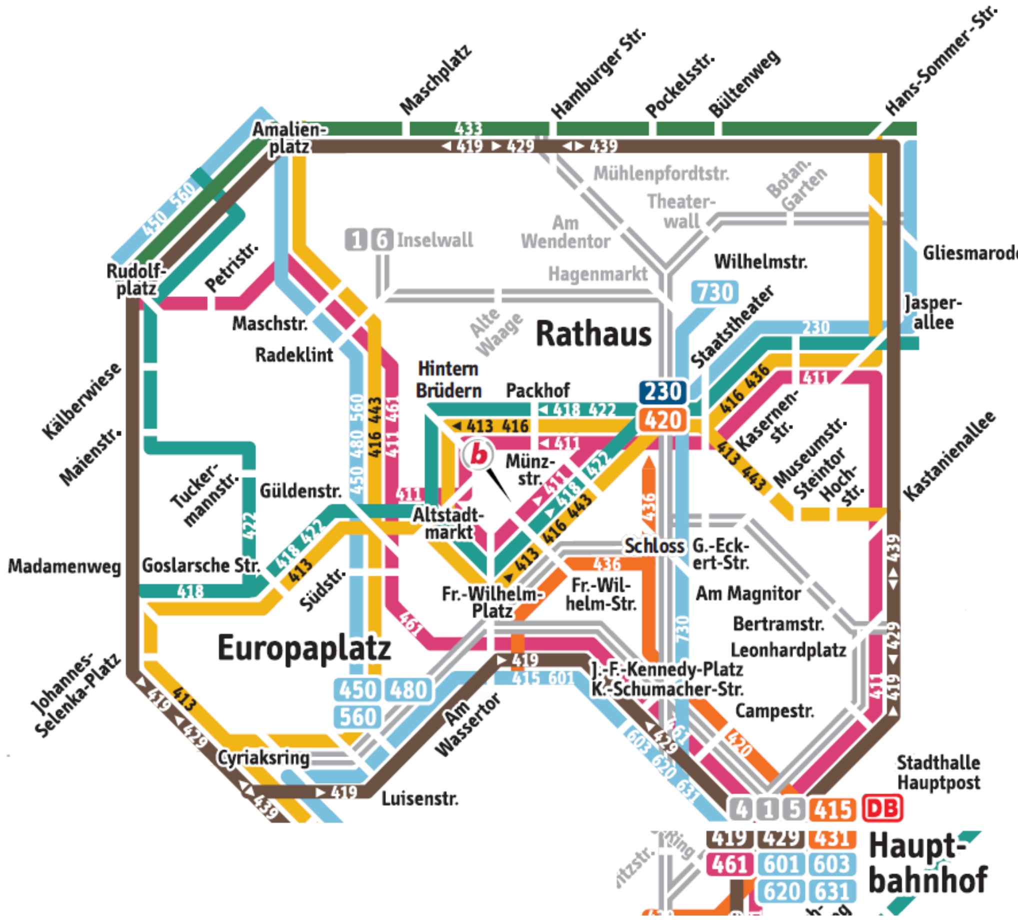

Figure 3.20: Public transport routes in Braunschweig modeled in the synthetic trace, image by operator [75]

created through a list of stops and each vehicle is assigned to one of these lines, and finally timetables are added. The result is a new movement file for the SUMO simulator. Third, SUMO generates mobility traces by microscopic simulation, i.e. traffic lights are considered, intersection-based right-of-way rules and different vehicle types. Thus, highly realistic mobility traces can be created, including disruptions in traffic flow. In the fourth step the movement files are converted for the DTN simulator ‘The ONE’[74]. The conversion adapts coordinate systems and matches the syntax of the mobility input reader of The ONE. After the last step a realistic simulation and comparison of the performance of various routing protocols in urban public transport is possible.

For the simulation, buses and light rail vehicles in the area of Braunschweig’s inner loop were modeled. The scenario shown in figure 3.20 has 54 stops and 13 lines which are defined by a series of stops. To each line one or several vehicles are assigned. Each vehicle serves a line and than stops for several minutes at the terminal stop, modeling the behavior of real buses/drivers that take a break before serving the same line again.

There are 28 vehicles simultaneously moving on the lines, and several additional off-duty vehicles taking a break at the terminal stops.

3.2.1.2 Seattle Trace

A trace [67] of roughly 1200 buses on 240 routes of the King County bus system was recorded in 2001 [65] and is available at the Crawdad[6] repository. The area around Seattle in which the buses are active is roughly 90 by 60 kilometers. Buses' positions are calculated by on-board Automatic Vehicle Location (AVL) systems. At the time that the trace was recorded AVLs installed in 1992 were in operation, these AVLs did not use GPS but a combination of signpost transmitters, odometry and mapmatching [76, 77]. Signpost based AVLs combined with odometry provide an accuracy of 1-20m [78], which is potentially better than GPS. The drawback is that these systems only track vehicles on fixed routes. This means that it is not possible to track off-route vehicles. Location updates of on-route vehicles are "polled irregularly but approximately every minute"[76] and the samples are "70% 12 minutes or less apart and 90% 20 minutes or less apart"[65].

The trace that is publicly available at Crawdad is not raw but already processed. According to the 'readme'-file[67] it contains timestamps, bus identifier, route identifier and coordinates. There is also a field that is specified as 'unknown'. Moreover, the coordinates "are in feet and were computed from the latitude and longitude values reported in the raw traces"[67], which are to the best of our knowledge not publicly available. There is also no additional metadata such as route definitions, stop positions and timetables available for the Seattle trace.

Figure 3.21 shows 10000 consecutive position updates from the Seattle trace. Each dot represents one data point. Routes are clearly visible because of the large number of samples plotted into the diagram. For visual comparison figure 3.22 shows a map (from Openstreetmaps) of the Seattle area. Note that Mercer Island is a good reference point. It is clearly discernible in figure 3.21 around coordinates (40000,20000).

The transformation and offsets between the coordinates of the Seattle traces and the commonly used WGS-84 system is not documented besides the statement that "coordinates are in feet and were computed from the latitude and longitude values reported in the raw traces" [67]. Unfortunately these raw traces are not available. Manual georeferencing implies the raw traces are state plane system coordinates (Washington State Plane North) with a proprietary offset subtracted. Nevertheless, we decided to keep both maps in their original format, since a conversion would be based on assumptions and manual georeferencing, and therefore unavoidably introduce inaccuracies.

3.2.1.3 Chicago Trace

Our Chicago mobility Trace [68] was recorded from the Chicago Transport Authority (CTA) Bus Tracker API, which is available and documented at [79]. Automated vehicle location systems (AVL) on the buses send position updates to a central server at CTA. These positions are based on GPS reception with a backup odometry system that is used if GPS reception is unavailable and also for plausibility checks (GPS receivers

[6]http://crawdad.cs.dartmouth.edu/

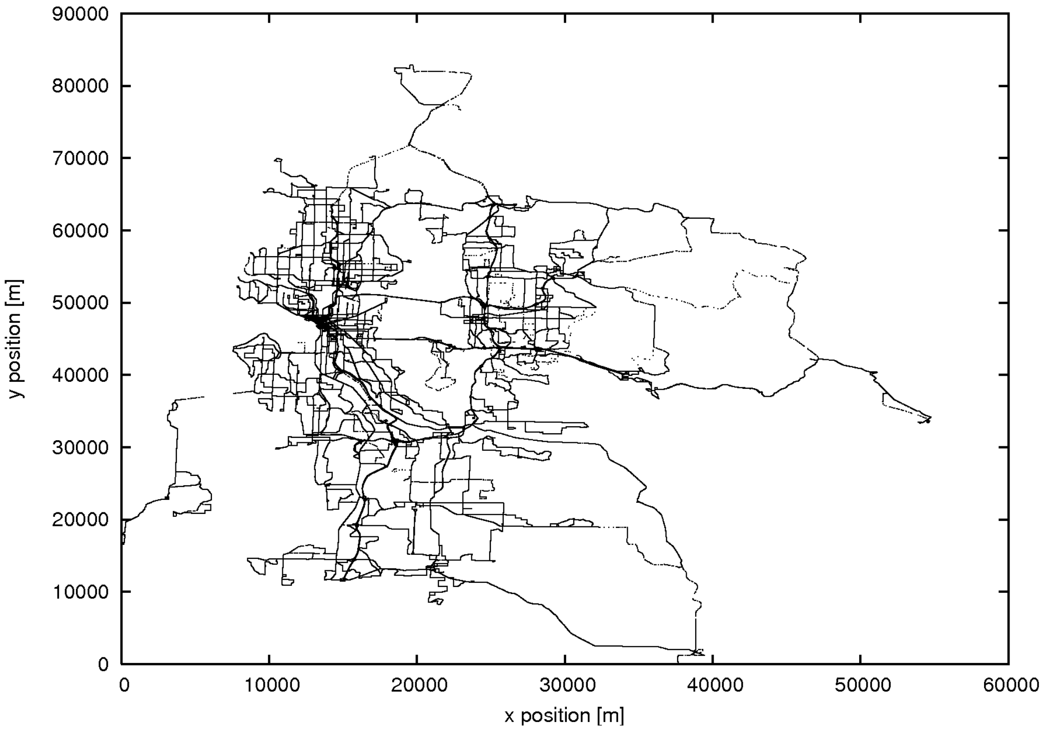

Figure 3.21: Position updates of the Seattle trace (aspect ratio optimized for printing).

occasionally report bogus positions if the reception is bad, e.g. in areas with many high buildings). For a continuous duration of 18 days in November 2009, a script was used to store timestamped vehicle identifiers and WGS-84[80] coordinates from the Bus Tracker API in a database. Moreover the trace contains the route and trip identifiers, direction and destination of the trip, and a pattern identifier as a geo-referenced representations of the trip. An advantage of the Chicago trace is the extensive meta-data that is available. It includes the names and geo-locations of stops as well as definitions of routes and timetables, which are publicly available via Google Transit[7] in a documented format[8]. This meta-data is essential for the development and evaluation of routing algorithms in a public transport network.

Figure 3.23 shows 10000 consecutive position updates from the Chicago trace. Again, routes are clearly visible because of the large number of samples. For this diagram the original unit of WGS-84 degrees of latitude/longitude were maintained. This standardized coordinate system is widely used, e.g. in geoinformation systems and map databases. For this reason the position updates and metadata (e.g. locations of busstops) can be processed without additional transformation, which often introduce inaccuracies. Therefore it is possible to display data to be analyzed as an overlay to cartographic material, e.g. like in figure 3.36. The number of active vehicles in the trace is plotted in Figure 3.24. Rush hours and weekends are clearly visible.

[7] http://www.google.com/intl/en/landing/transit/text.html
[8] http://code.google.com/transit/spec/transit_feed_specification.html

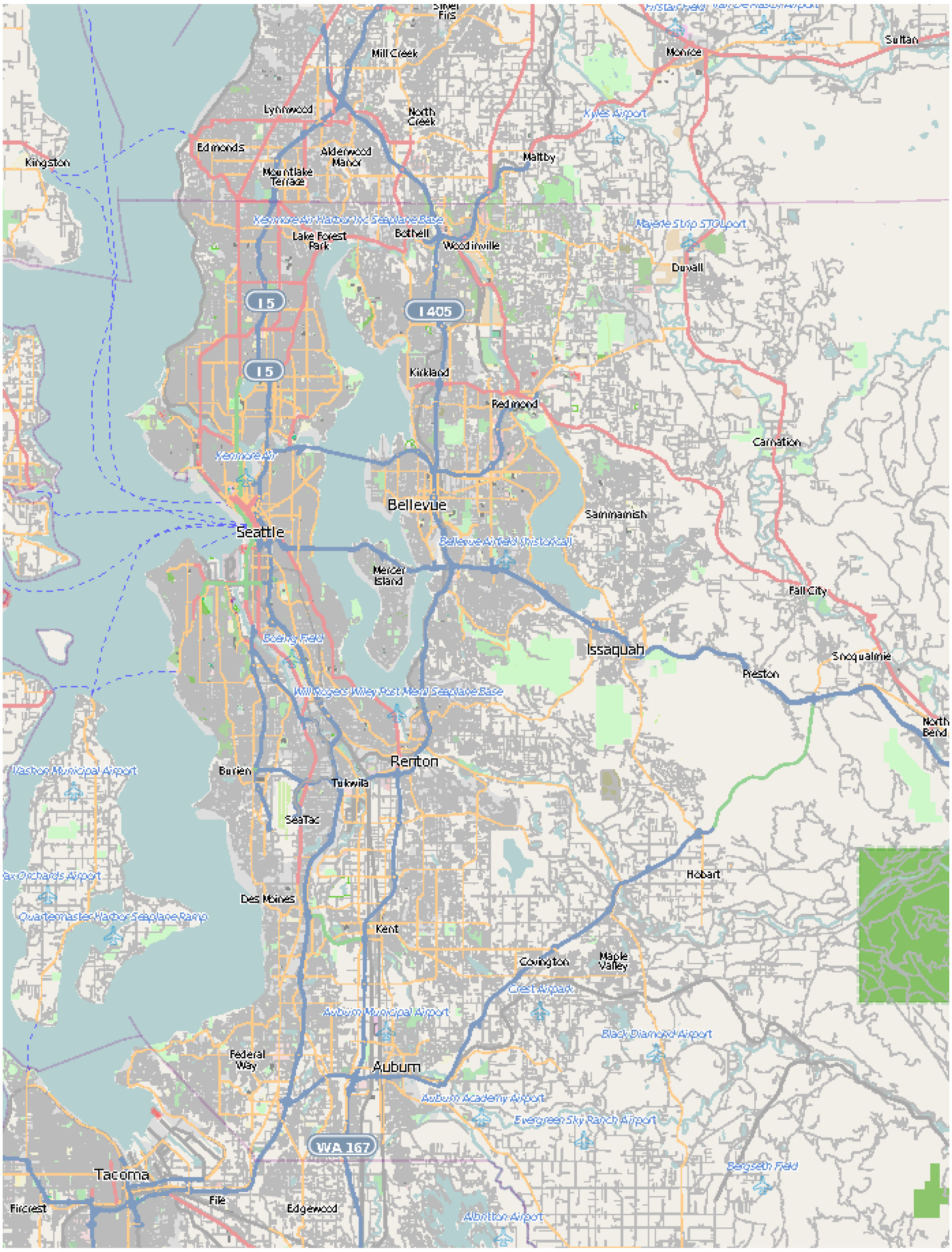

Figure 3.22: Map of the Seattle area (provided by OpenStreetMaps) as a visual reference. Note that Mercers Island in the middle of the map is also clearly visible in the plot of position updates in the previous figure.

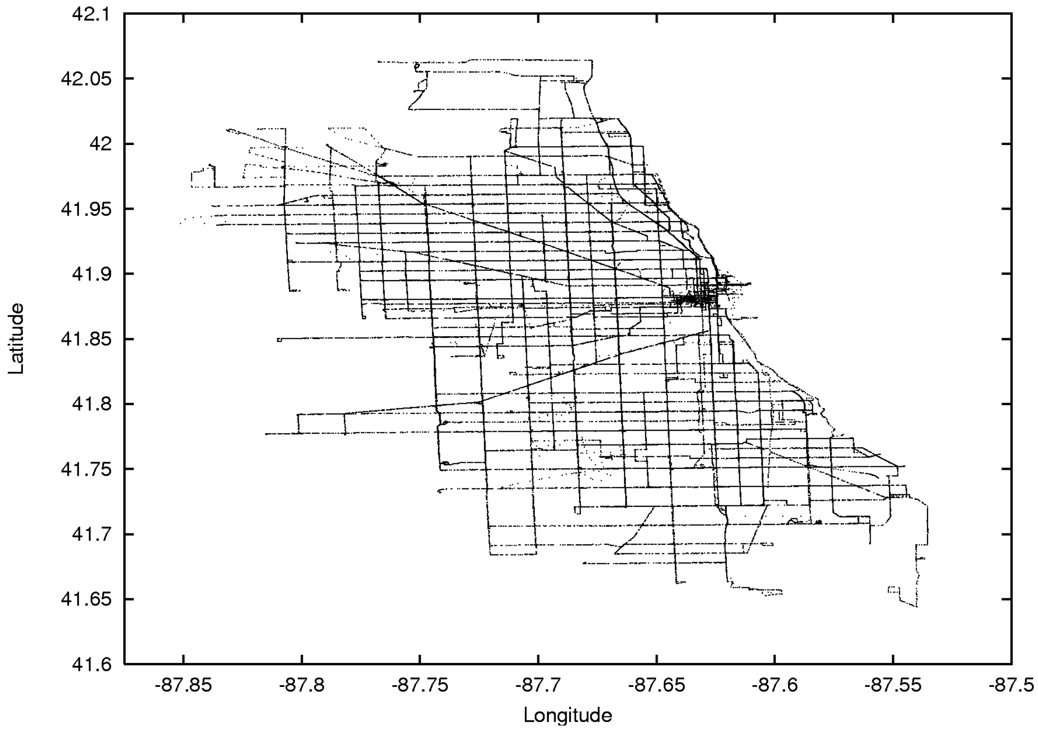

Figure 3.23: Position updates of the Chicago trace (aspect ratio optimized for printing).

Arrival Times and Delays There are two basic types of service level definitions for timetables in public transport systems. The most common type is to define schedules for every stop with exact arrival times of each vehicle trip. This gives customers the best service, since they are able to reliably plan their departure times, which helps to prevent annoying waiting times. On the other hand, this type of schedule is harder to abide and delays have a severe influence on service quality. The other option is to define service slots with time intervals. As an example, the schedule of line 21 in the Chicago Trace defines that starting at 8:07 on Monday till Friday a bus will arrive "every 14 to 16 minutes"[9]. From a customers point of view this more inconvenient than a fixed schedule, but at least he can count on defined average and worst case waiting times. For the operator, this kind of schedule is much more flexible. In the case of a traffic jam it is sufficient to dispatch additional vehicles on the according line to fulfill a schedule. Therefore, this kind of schedule is suitable for public transport lines on which disturbances such as traffic jams are very common.

Now an analysis of the arrival times on such lines will be given. Traffic disturbances are rather common in Chicago during rush hours and therefore interval based timetables are partially used in Chicago. As a representative example of lines that have interval

[9] http://www.transitchicago.com/asset.aspx?AssetId=843

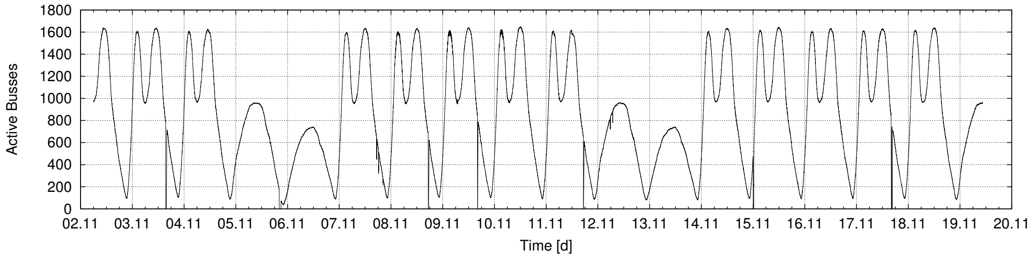

Figure 3.24: Number of active buses over 18 days of the trace, with clearly visible rush hours and weekends.

timetables during rush hours the lines 21 and 8 are used. In the grid-like road network of Chicago, line 21 operates west-to-east and line 8 north-to-south and vice versa. Both lines cross at the intersection of Cermak and Halsted. The stops of line 21 are named "Cermak+Halsted" and are both located on the west of the intersection. Line 8's stops are named "Halsted+Cermak" and located north of the intersection for the northbound direction, and south for the southbound direction.

The inter-arrival times are plotted in figures 3.25 and 3.26. For the analysis a time window from 8:30 till 17:30 of three weekdays was chosen. During this time the interval timetable is effective. Therefore, the expected result at on-time arrivals would be a peak in the area between 14 and 16 minutes (840-960 s) for line 21. In fact, there is a significant number of inter-arrival times observable within this interval in figure 3.25. For line 8, there is an asymmetrical interval schedule defined. It is 6-12 minutes for the northbound direction and 9-12 minutes southbound. Again, this is consistent with the data in figure 3.26. Note the peak that is shifted slightly to the right for the southbound direction and the flatter maximum for northbound. However, in both figures it is also obvious that there are many arrivals outside of the interval defined by the timetables. There are significant numbers of late and early arrivals on both lines. Early arrivals are usually caused by additional vehicles that are put into service because of high demand or as an reaction on disturbances. Late arrivals are usually caused by road traffic conditions.

In conclusion the plotted inter-arrival times show that there are deviations, but also that a large number of arrivals are within the planned interval. Unfortunately, this analysis is only possible with the Chicago trace, because the Seattle trace lacks the important meta-data (e.g. locations of bus-stops, mapping of lines to vehicle IDs, timetables). Therefore, a comparison between both traces is not possible. However, a quantitative description of the subjectively obvious fact that urban public transport vehicles are not always on time was given for the Chicago trace. For the design of a time-table based routing the significant number of deviations is important information. A routing decision should better not rely on buses to be perfectly on time. Moreover, routes may need to be re-planned if a too high deviation occurs and planned contacts are not happening as predicted.

Position Data Quality The position data in the Chicago trace is transmitted by on-board Automatic Vehicle Location (AVL) systems to the central backend that provides

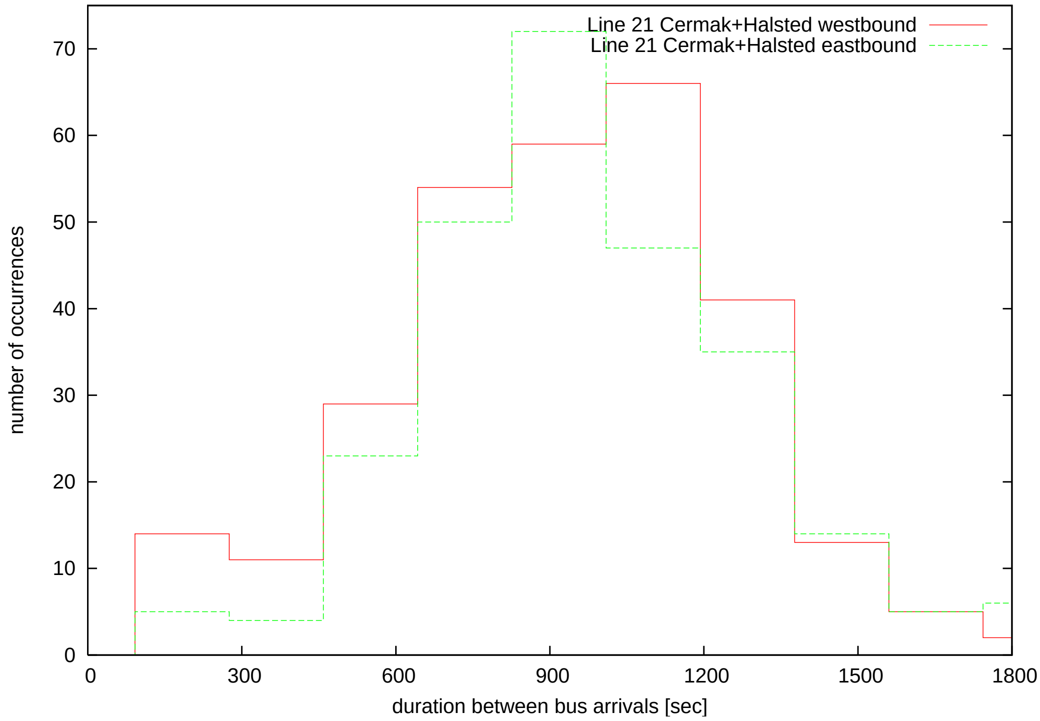

Figure 3.25: Distribution of inter-arrival times of line 21 at the stop "Cermak+Halsted".

a web service interface that was used to record the trace. The AVL systems receive GPS positions. Occasionally, the GPS reception is not good enough so that the calculated position is not reliable or inaccurate. In urban environments this is usually caused by high buildings or other structures that cause signal attenuation or multipath propagation. This is a usual, system inherent behavior of GPS and its details are beyond the scope of this thesis. The quality of GPS reception has a relevance for any GPS-based trace. However, there is an additional special property in the Chicago trace: in the case of bad receptions the AVL systems use additional on board systems e.g. odometric systems that register distance and an electronic compass calculate (extrapolate) positions. Unfortunately, these systems that continuously extrapolate the new position from an old position are rather inaccurate. Moreover, the GPS receiver sometimes generates a very inaccurate but valid position before reception is lost. Then the odometry/compass-system extrapolates positions until the GPS reception gets good enough again. This leads to positions traces that seam to "jump". Such a behavior is shown in figure 3.27. A public transport vehicle operates on a line that is mainly in a north-to-south orientation. The AVL system first receives valid GPS positions, which can be observed by the positions along the north-south road. Then the reception gets worse, and the GPS receiver generates a very inaccurate position far to the west of the vehicles route. This is when the first "jump" occurs. Now GPS no longer reports position data and the odometry/compass backup takes over and creates a series of plausible positions that are initialized with the last (unfortunately very inaccurate) GPS position. Within the trace it now seams that

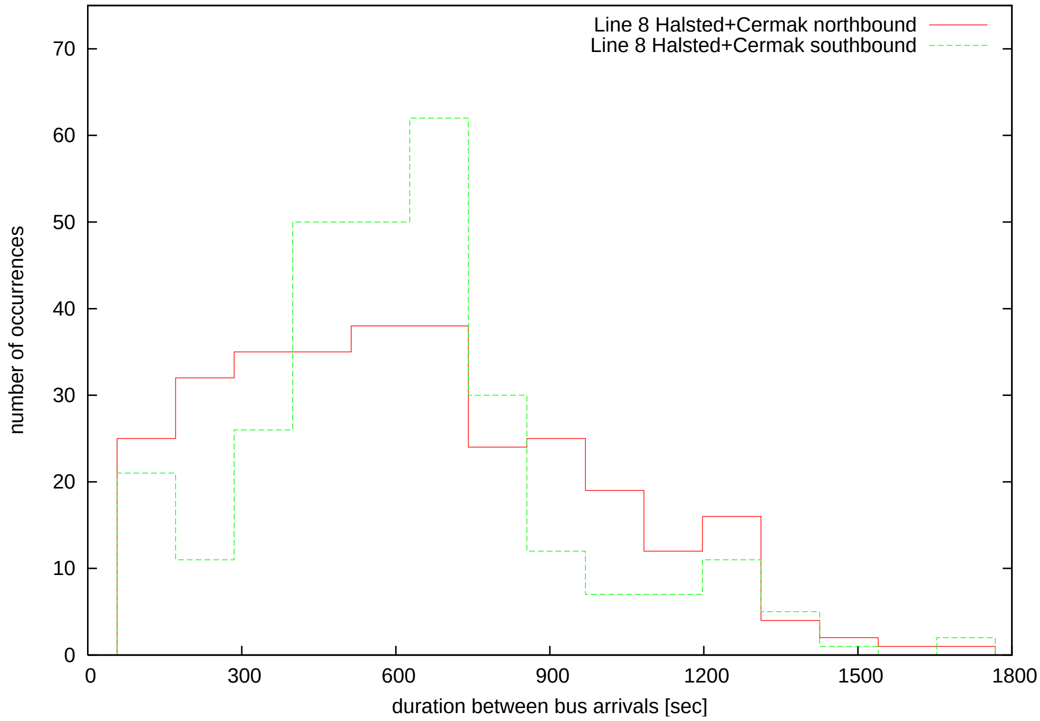

Figure 3.26: Distribution of inter-arrival times of line 8 at the stop “Halsted+Cermak”.

the vehicle drives to the north-east. In this extreme example a comparison to the map shown in figure 3.27 shows that this data is obviously wrong. Some time later the GPS reception gets better and now valid GPS positions are reported again. This is when the second “jump” back to the vehicles route seams to occur.

It should be noted that the worst-case example in figure 3.27 is not representative and that only about 0.03% of the reported positions show these jumps. Moreover, it is relatively simple to detect jumps by checking for sudden implausible changes in vehicle velocity.

However, this behavior shows the drawback of using real-world data instead of synthetic traces for evaluation. Since the data is not “perfect”, certain decisions must be made in order to deal with these imperfections. In this case the options are dropping the complete vehicle from the trace or just dropping the invalid positions between the jumps and repairing the trace with extrapolated positions between the last and first valid GPS position. However, the last option would have led to even more assumptions being made in the later evaluation. Therefore, and because the number of “jumps” is very low, we decided to completely eliminate jumping vehicles (8 vehicles, which is lesser than 0,5%) from the traces that are used for simulation based evaluation. It can be argued that dropping vehicles will lead to a seemingly worse performance. On the other hand “repaired jumps” may lead to a seemingly better performance. We decided for the first option in order to prevent any overenthusiastic simulation results. It also has to be noted that this does not lead to any “unfair” simulation results, since the same trace is used

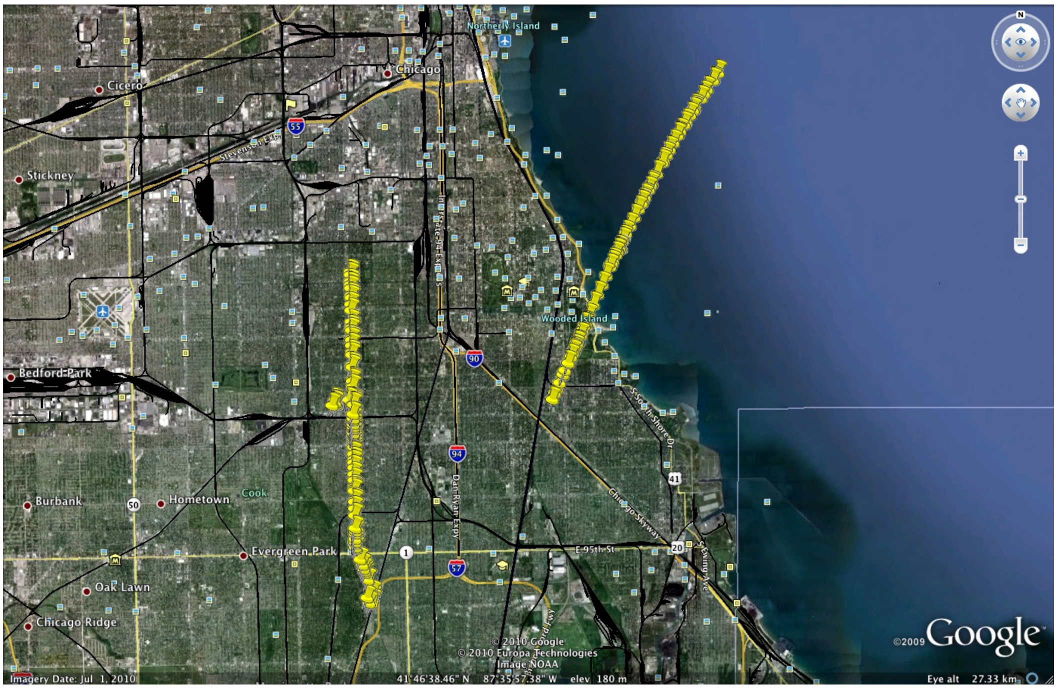

Figure 3.27: AVL systems occasionally report wrong position data when the GPS reception is bad and the backup odometry/compass system is used. Picture by Google.

for all simulation scenarios.

3.2.2 Seattle and Chicago Traces: Similarities and Differences

3.2.2.1 Amount and Interval of Position Updates

The interval of position updates is important for the resolution of a trace. The more updates (samples) are sent by the AVL systems in a given period of time the better the resolution. Unfortunately, the update rate is bounded, since sending updates requires usage of limited resources (e.g. usage of the shared voice/data radio system to send position data). For simulations, it is necessary to create continuous node movement by extrapolating [66] the discrete position updates. Therefore, it is desirable to have short update intervals.

Comparing the 2001 traces of Seattle [65] with our 2009 traces of Chicago shows several similarities and differences. In order to achieve a fair comparison a Monday workday in November is selected from both traces. During this 24 hour period the vehicles in Seattle sent 327880 valid position updates and those in Chicago sent 1736431, more than five times as many. The number of active vehicles peaks to 1647 in Chicago and 765 in Seattle, as can be observed in Figure 3.29. Both traces show similar distinct rush hour spikes and vehicles move mainly at low speeds under 35km/h. However, the

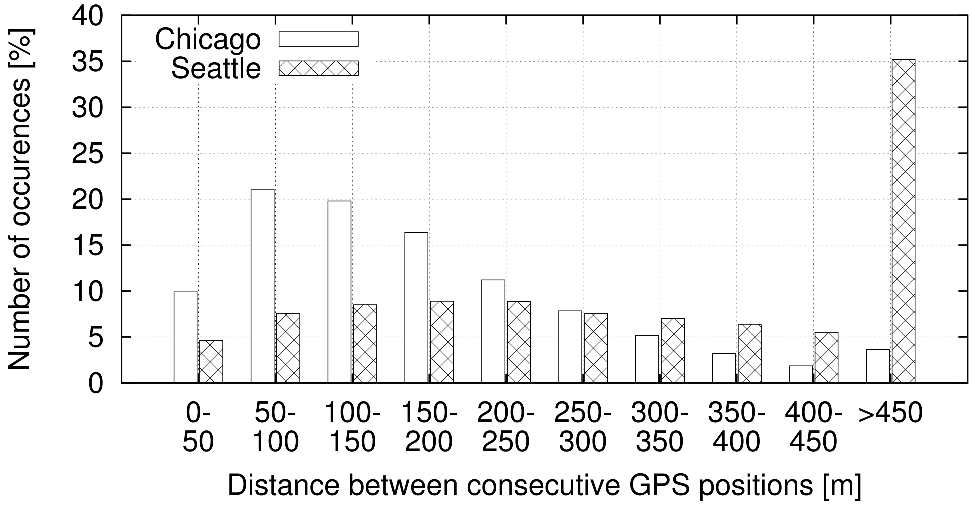

Figure 3.28: Distances between GPS updates.

speed distribution plotted in Figure 3.30 shows that vehicles drive more often at higher speeds in Seattle, which is due to a more rural area of operation in contrast to the denser downtown traffic in Chicago. Further, the update interval shown in Figure 3.31 is shorter in Chicago, leading to a larger number of position updates. The rate is mainly between 20 and 40 seconds in Chicago but in the order of 1-2 minutes for the Seattle trace. Therefore, the Chicago trace is better suited as a basis for generating realistic mobility traces for simulations. It contains more detailed, well-grounded base data compared to the Seattle trace. Due to the lack of supporting points in the latter, more extrapolations are needed which lower the quality of simulations and other studies based on the Seattle data.

3.2.2.2 Vehicle-to-Line Assignment

The existence of lines (i.e. predefined routes) is an important property of public transport networks. Moreover, lines describe future vehicle trajectories and therefore offer interesting opportunities for ad-hoc and delay tolerant routing algorithms. For this reason it is important to understand the dynamic relationship between specific vehicles and assigned lines. Usually there is no fixed assignment of vehicles to routes, because each vehicle is dispatched to a random route at the beginning of a shift. [64] Moreover, during one shift vehicles frequently change routes in order to increase usage rate by reducing deadheads and waiting times.

An analysis of the Chicago vehicle-to-line assignment shows a non-deterministic behavior similar to that described in [64]. Moreover, it is also observable that a significant number of vehicles changes from one line to another during a shift. In figure 3.32 each bar shows the number of vehicles (y-axis) and the number of distinct lines (x-axis) they

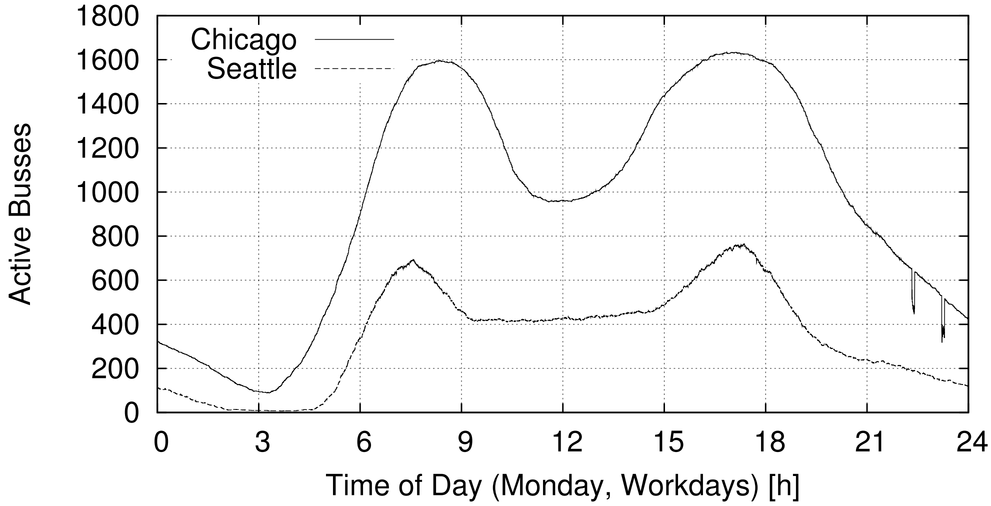

Figure 3.29: Number of active buses on Mondays.

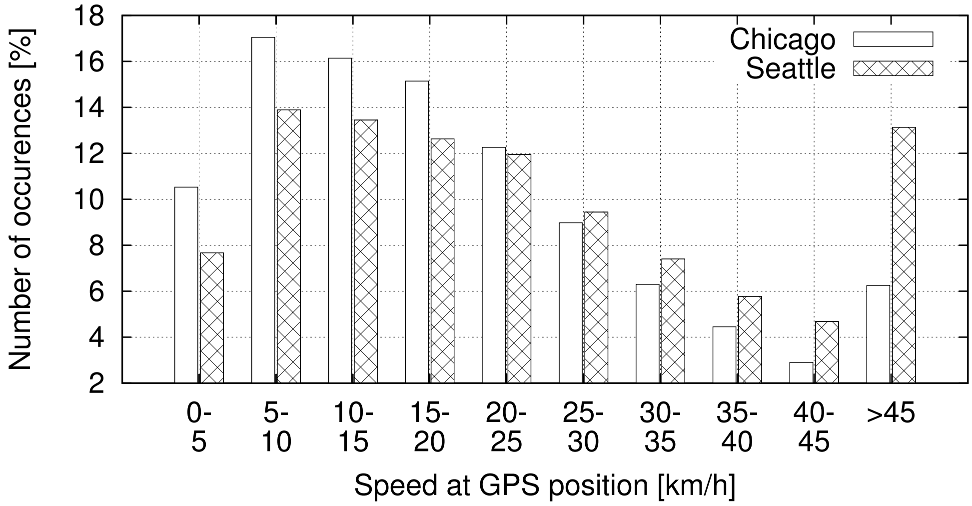

Figure 3.30: Distribution of Vehicle speeds.

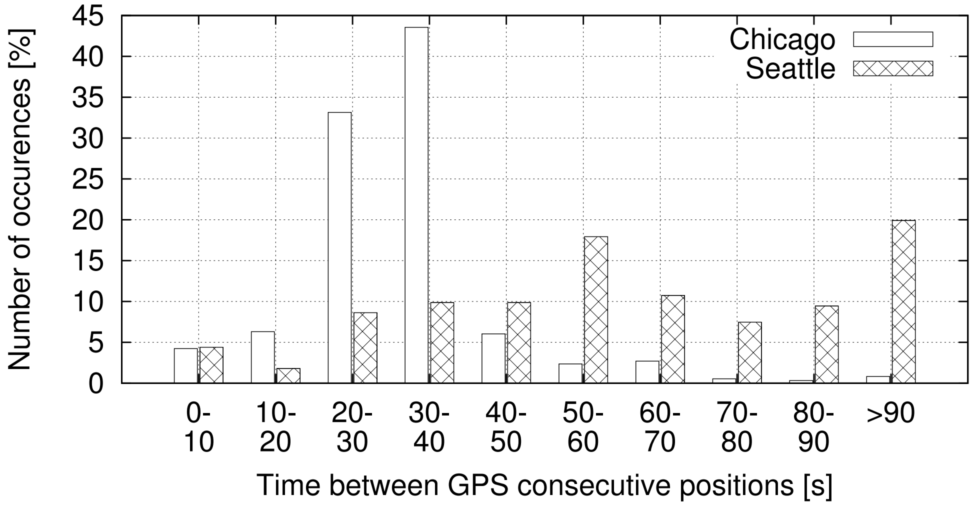

Figure 3.31: Distribution of time between consecutive position updates.

operate on, during the Monday morning five hour time window described above. This means that only 440 of 1688 vehicles operate on the same line during these five hours, while the remaining vehicles change lines one or more times. Figure 3.33 shows the number of vehicles/lines for the whole trace (18 days). It is observable that the majority of vehicles operates on 15 or more lines.

The Seattle trace shows a similar behavior, although with a different markedness. In a five hour window (workday Monday morning 7-12h, equivalent to Chicago) shown in figure 3.34 there are much fewer vehicles changing lines. Moreover, changes are less frequent, but nevertheless not uncommon. Figure 3.35 shows the vehicle to line assignment over 16 consecutive days. Interestingly, in Seattle there are slightly more than 40 vehicles that exclusively operate on a single line, which is significantly more than in Chicago. However, this exclusive operation is still very uncommon considering the large number of vehicles that change lines.

3.2.2.3 Vehicle Density

The density of vehicles significantly impacts the performance of vehicular ad hoc networks (VANETs) and DTNs. Moreover, knowledge on density distribution of vehicles with specific properties is valuable for the design of routing and scheduling algorithms. The GPS position updates of the Chicago trace provide density information. Figure 3.36 shows a snapshot of vehicle positions plotted on a map of the Chicago area. There is a clear accumulation of vehicles at the city center.

For a more detailed analysis a grid of 0.001° was laid over the WGS-84 coordinates of the position updates. Then the number of position updates for each resulting tile was extracted from the trace database for specific time windows. The number of updates within each tile divided by the number of all updates corresponds to the probability to encounter a vehicle in that tile, because of the fixed time intervals between position

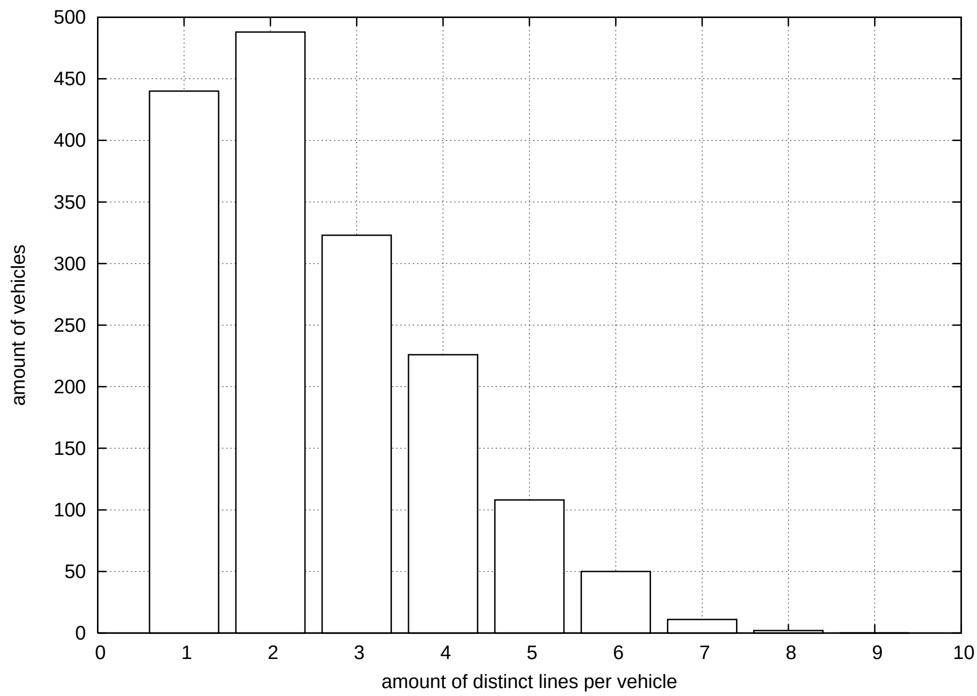

Figure 3.32: Chicago vehicle to line assignment for a five hour time window (Monday morning 7h-12h)

updates. Figure 3.37 shows the number of updates per tile for a five hour time window. Interestingly, the distribution looks different than what one would intuitively expect on the basis of the snapshot in figure 3.36. Besides the city center, there are other areas with high amounts of updates. This happens because slow and non-moving vehicles stay longer in a tile and therefore send more updates. Especially the bus garages and the bus-stops where off-duty vehicles/drivers take their break show this accumulation. This means that there is a higher probability to encounter a vehicle in such places. However, these vehicles are not moving for certain periods, so it cannot be assumed that these areas are generally "better" for the performance of VANETs/DTNs.

In the next step of the analysis, vehicles operating on the same line are isolated. Figure 3.38 shows the density of updates for line 8. Not surprisingly the 'footprint' of line 8 is clearly visible, but again there are tiles with a significantly higher encounter probability. As another example line 21 is plotted in figure 3.39. It has a similar distribution, but another orientation. In conclusion the encounter probability is not evenly distributed on a single line.

The density data plotted in the diagrams has for example the potential to be exploited for a routing approach, because grid datastructures are relatively easy to compute. For example, the density of moving vehicles on different lines can be used to identify tiles in which there is a large probability that vehicles of different lines encounter each other.

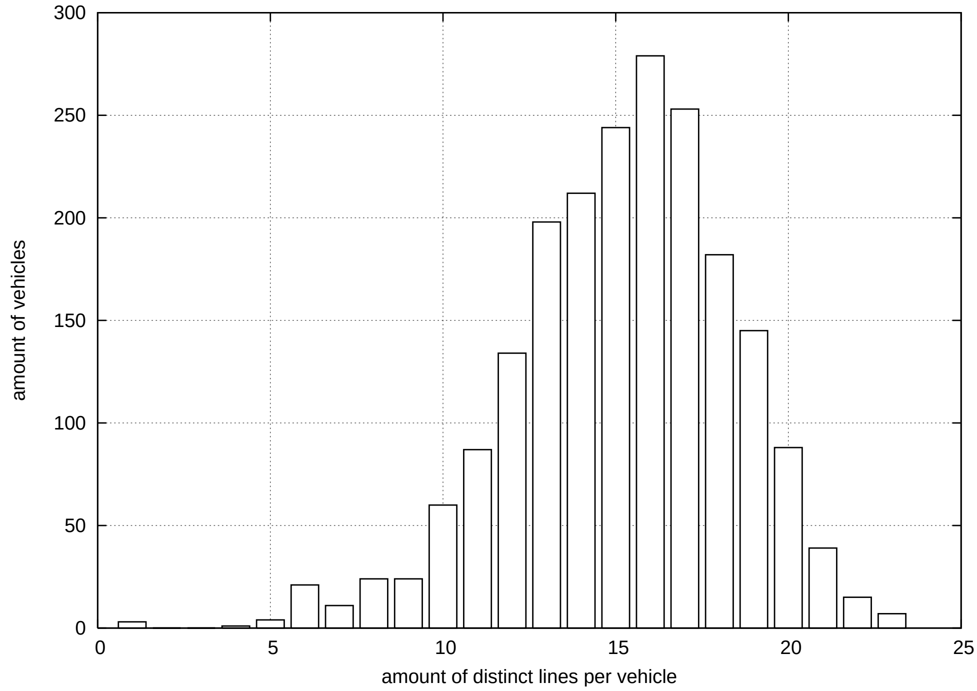

Figure 3.33: Chicago vehicle to line assignment for the whole duration of the trace (18 days)

Figures 3.40 and 3.41 show the spatial distribution of 10.000 consecutive position updates in the Seattle trace. The distribution of updates directly corresponds to the distribution of vehicles, since updates are transmitted at regular intervals. It is highly uneven as the normalized number of updates in tiles of 100m×100m in figure 3.40 depicts. The big agglomeration is located at the downtown area and can be explained by the denser public transport network in this area. The suburban and rural areas are much sparser. For this reason the downtown area appears as a big spike in the plot. In figure 3.41 the downtown area is zoomed in by choosing a smaller range of easting/northing (note the coordinates in the diagrams to observe the zoomed-in area). The close-up shows that the single spike is in fact more differentiated. Now it is possible to recognize longish patterns, which are in fact frequently used roads.

3.2.2.4 Comparison Summary

Table 3.1 summarizes the similarities as well as the differences of both large scale movement traces. Both traces were recorded over a duration of more than two weeks, which is sufficient to analyze weekly dynamics such as working day and weekend cycles. The maximum number of active vehicles (i.e. on-duty vehicles with active AVL systems) in the Chicago trace is more than twice as high as in Seattle. It is important to note that the covered areas of both public transport systems are very different, not only in size but also in density and road traffic properties (e.g. inner city vs. urban). This

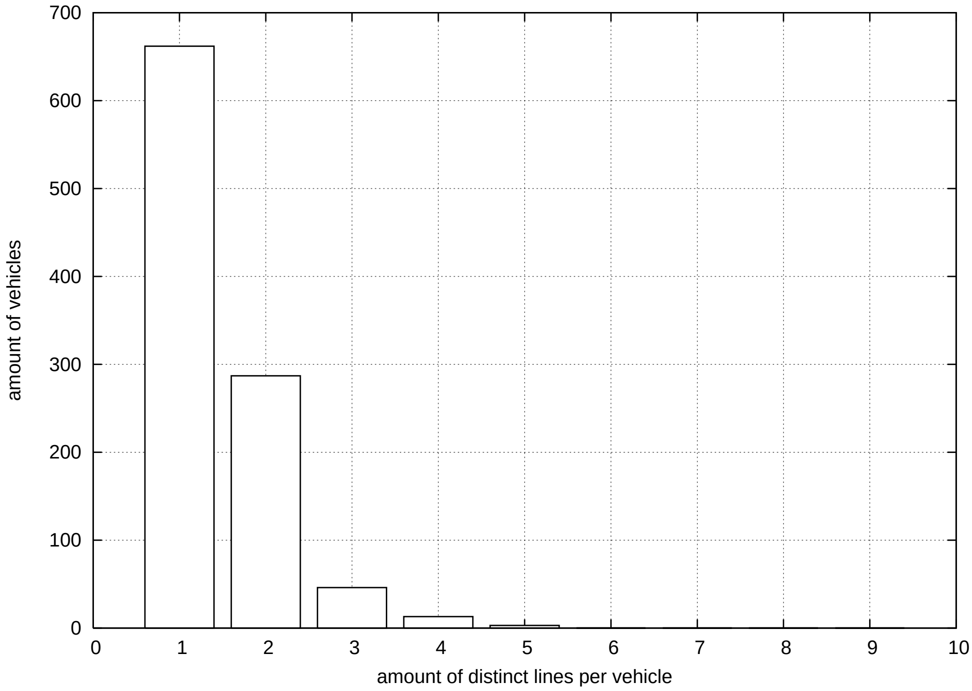

Figure 3.34: Seattle vehicle to line assignment for a five hour time window (Monday morning 7h-12h)

difference is also clearly visible in the three-dimensional spatial distribution plots and by the slightly higher vehicle speeds in the Seattle area. Moreover, it also becomes obvious by the difference in vehicle-to-line assignment that the public transport systems are operated in slightly diverse way. Unfortunately, the traces were recorded with different coordinate systems, which is induced by the position data generated by different kinds of proprietary AVL systems. Therefore, there is also a dissimilar extent of data fields in the traces. While the Seattle trace contains only basic data, there is a rich set of additional meta-data available for the Chicago trace. Moreover, the position update interval of the AVL systems used in Chicago is significantly shorter. For this reason (and also because of the larger number of active vehicles in Chicago) there are more than five times as many updates in the Chicago trace. However, although there are many differences, there are also similarities in the dynamics of mobility. Very distinct rush-hour spikes as well as weekday/weekend patterns are present in both traces, and the number of active vehicles over time follows a very similar pattern.

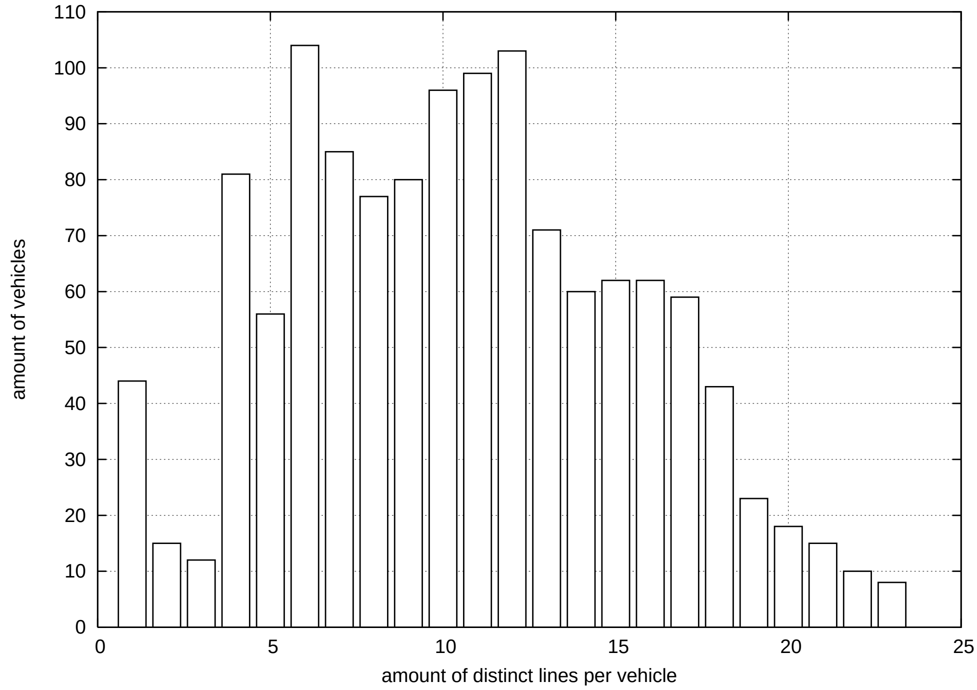

Figure 3.35: Seattle vehicle to line assignment for the whole duration of the trace (16 days)

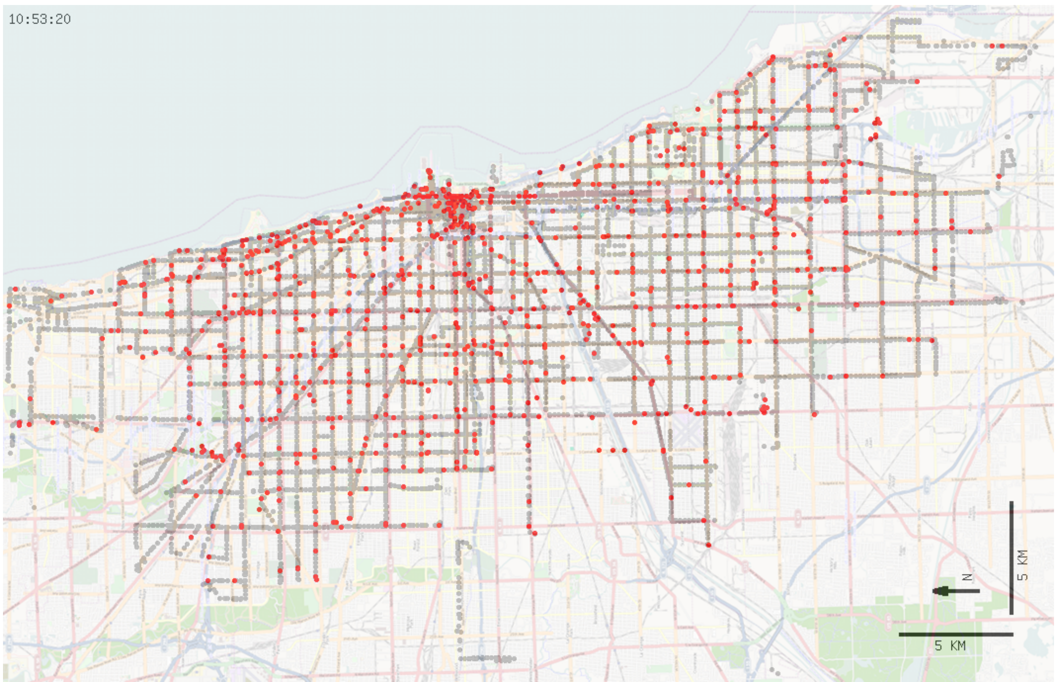

Figure 3.36: Snapshot of vehicle density. Red dots indicate vehicles, gray dots indicate stops. The background map is based on OpenStreetMaps.

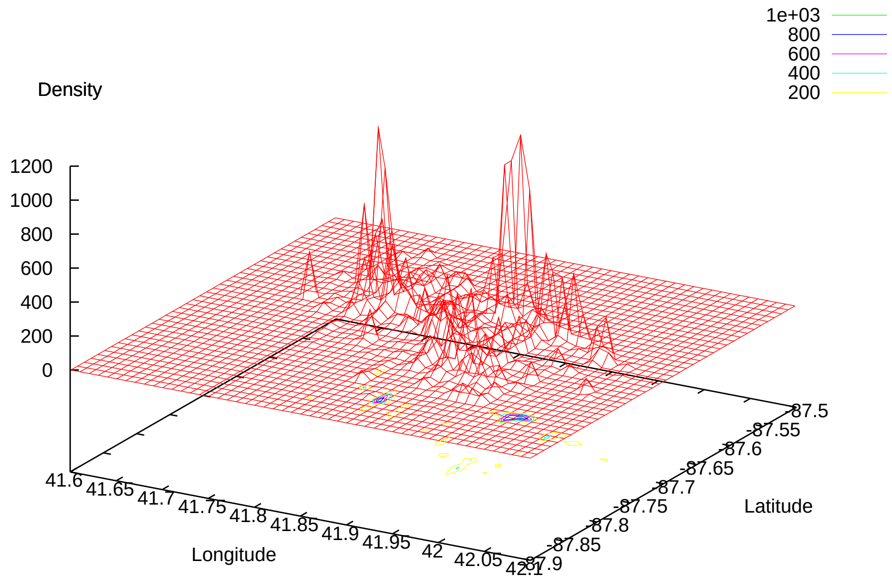

Figure 3.37: Chicago trace: Density of position updates (corresponds to vehicle density) for a five hour time window (Monday morning 7h-12h)

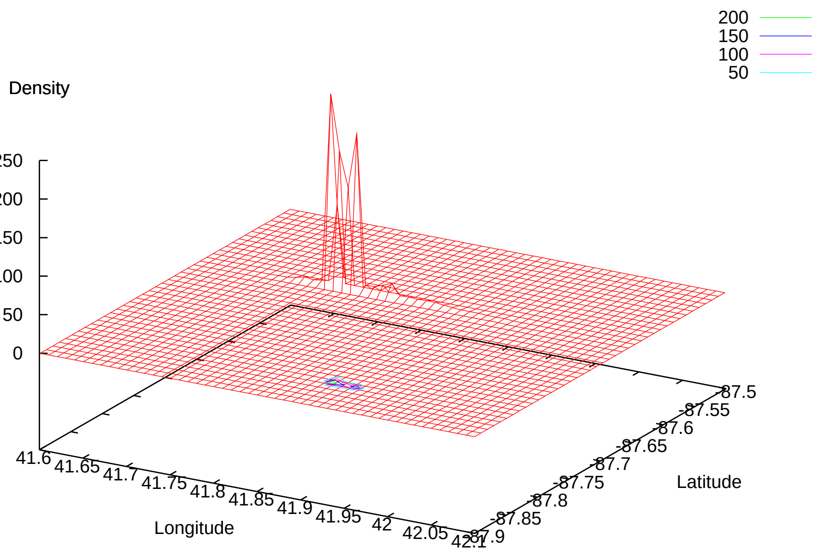

Figure 3.38: Chicago trace: Density of position updates on line 8 for a five hour time window (Monday morning 7h-12h)

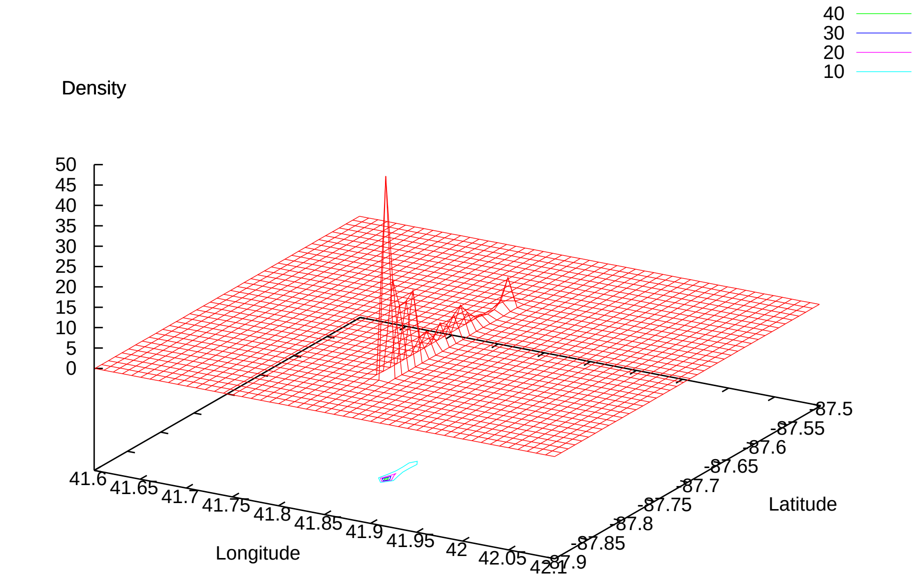

Figure 3.39: Chicago trace: Density of position updates on line 21 for a five hour time window (Monday morning 7h-12h)

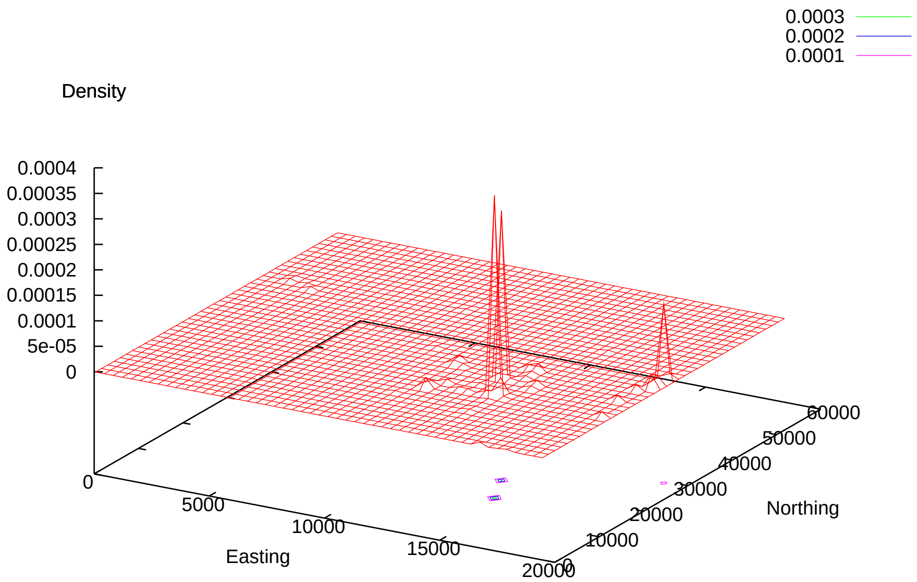

Figure 3.40: Seattle trace: Normalized spatial density of position updates.

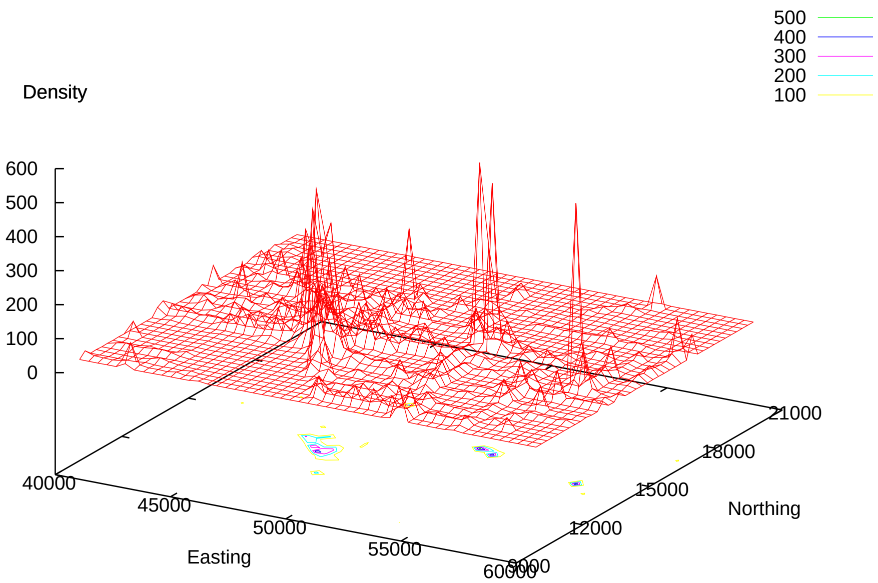

Figure 3.41: Seattle trace: Spatial density of position updates within the city area.

	Seattle 'Trace5' [67]	Chicago [68]
duration	16 days	18 days
max. number of active vehicles	765	1647
covered area (enclosing rectangle)	≈ 30×25 km	≈ 55×85 km
coordinate system	Washington State Plane North	WGS 84
AVL system type	signpost + odometry	GPS + odometry backup
trace data	timestamp, position, route	timestamp, position, route, trip, direction, destination
available metadata	-	timetables, stops, route and trip definitions
position update interval	26-120s	20-40s
number of updates (5h)	327 880	1 736 431
lines per vehicle, min/avg/max (5h)	1 / 1.4 / 5	1 / 2.6 / 8

Table 3.1: Overview about similarities and differences.

3.3 Summary

In this chapter the basic communication characteristics and properties were examined. It was shown that both IEEE 802.11b and 802.11a are well suited and economically promising technologies. Both have been evaluated in real world scenarios to obtain realistic performance parameters that will be required for a sound system design.

Further, this chapter introduced several mobility traces. First, the generative Braunschweig Model, which is used to create a synthetic but realistic trace. This allows for a controlled environment, which has advantages for the development and testing of routing algorithms, because specific situations (e.g. delays due to traffic jams) can be generated. On the other hand such a trace can only contain known and expected properties. Therefore, real traces are required.

The second trace (Seattle) that was introduced, originates from a real bus-based transport network, but lacks important context information such as route descriptions and timetables. Moreover, its position accuracy is not very high because the AVL system is sign-post based. However, a comparison of the Seattle trace to the third trace (Chicago) has shown that there are similar properties (e.g. rush hour spikes) and interesting effects that were rather unexpected. For example, knowledge of the changing vehicle-to-line assignment and the high inter-arrival variance is very important for the design of context-aware routing algorithms. Not knowing these properties and only relying on a synthetic trace would result in a “broken-by-design” algorithm, which would show a great performance in the simulator but fail in a real vehicular DTN.

Chapter 4

Contact Analysis

4.1 Characteristics of Contacts

In the previous section the basic suitability of WLAN for vehicular DTNs was examined. Now that the radio channel characteristics such as range and datarate have been evaluated, a closer look at the properties of a communication contact between two vehicles is necessary. These contacts are mainly influenced by the radio channel characteristics and the vehicles mobility pattern that was discussed in the previous chapter. There are several indicators that can be used to estimate how "challenging" a scenario or mobility model is with regard to DTN routing, and to give an indication of the expected performance:

Inter-contact time is defined as the time that passes between successive contacts of two given nodes. In other words inter-contact time starts to count at the end of a contact and stops at the beginning of the next contact of exactly the same pair of nodes. Inter-contact time is often used to characterize human mobility or rather mobility of human carried devices [81, 82, 83]. Low inter-contact times indicate pairs of well connected nodes with a low latency. For direct delivery schemes this allows for a valid and intuitive evaluation. But for the expected performance of routing schemes in complex scenarios the average inter-contact time may provide a rough estimation, but only in combination with other metrics (e.g. number of contacts in proportion to the number of nodes).

Any-inter-contact time is similar to inter-contact time, but only one node is given, instead of a pair of nodes. Therefore, any-inter-contact time starts to count at the end of a contact and stops at the beginning of a contact between exactly the same node and any other node. In [84] any-inter-contact time was used to characterize the mobility of trains. Any-inter-contact time is a metric that describes how "well-connected" a given node is to other nodes. For complex scenarios a high proportion of nodes with low any-inter-contact times may indicate potential for good performance of DTN routing.

Contact duration [1] is the time that passes between the beginning and the ending of a contact between a given pair of nodes. In bus-based delay tolerant networks the duration of a contact between two nodes limits the amount of data that can be

[1] sometimes referred to as simply "contact time", but this term will not be used here because it is too ambiguous.

transferred. For the implementation of such networks it is important to determine which contact durations can be expected. Therefore, in this work we use contact duration as the main metric.

4.1.1 Contact Analysis

In this section different types of contacts are structured and the effect of radio range on contact duration are examined and characterized. For realistic results we build on the experimentally investigated communication characteristics and our real mobility trace, which were both described in the previous chapter. We report the simulation results for different types of contacts and the probability distribution of contact durations for various realistic ranges. Furthermore, we show that the angle of contacts is an appropriate criterion for the classification of contacts, and propose to use it as input for routing and scheduling decisions.

The goal of this section is to characterize types of contacts and the effect of range on contact duration, to gain new insights for the system design and for routing algorithms in real-word public transport DTNs. The approach to obtain realistic results is to use our mobility trace of a real large-scale public transport network.

Prior work on mobility properties [84] and mobility modeling [64] of bus-based DTNs focuses on inter-contact time. In this work we analyze another important metric - the duration of contacts. Contact duration has a direct impact on the amount of data that can be transferred during one encounter. In [85] this amount has been experimentally evaluated for a vehicle which drives past a WLAN access point on a highway. However, these results cannot be applied to bus-based DTNs, in which a wireless connection is established between two moving vehicles when they get within each others radio range (a situation that is more complex than with one vehicle and one stationary access point). The connection then remains available until the distance between the moving vehicles exceeds the radio range. Distance, range and vehicle speed are the main impact factors on contact duration, as we investigated in [17]. Distance and speed are dictated by road and traffic conditions, these variables are complex and hard to influence during system design. In contrast, range can be influenced by choosing the wireless technology, hardware (e.g. radio, antenna) and software (e.g. transmit power control algorithms implemented in firmware or driver).

4.1.1.1 Basic Contact Types

In a bus-based DTN, vehicles move on predetermined routes which result in different contact types between vehicles as shown in figures 4.1 to 4.3.

1. **Encounter**: Two vehicles pass each other in opposite directions (see figure 4.1).

2. **Intersection**: Two vehicles pass each other in orthogonal directions (see figure 4.2).

3. **Following**: One vehicle follows another over a certain path length (see figure 4.3).

4. **Not Moving**: One vehicle passes by a vehicle which does not move.

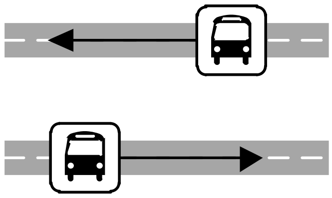

Figure 4.1: Encounter

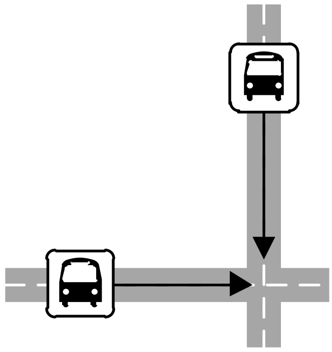

Figure 4.2: Intersection

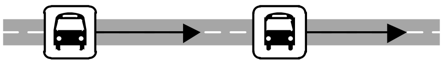

Figure 4.3: Following

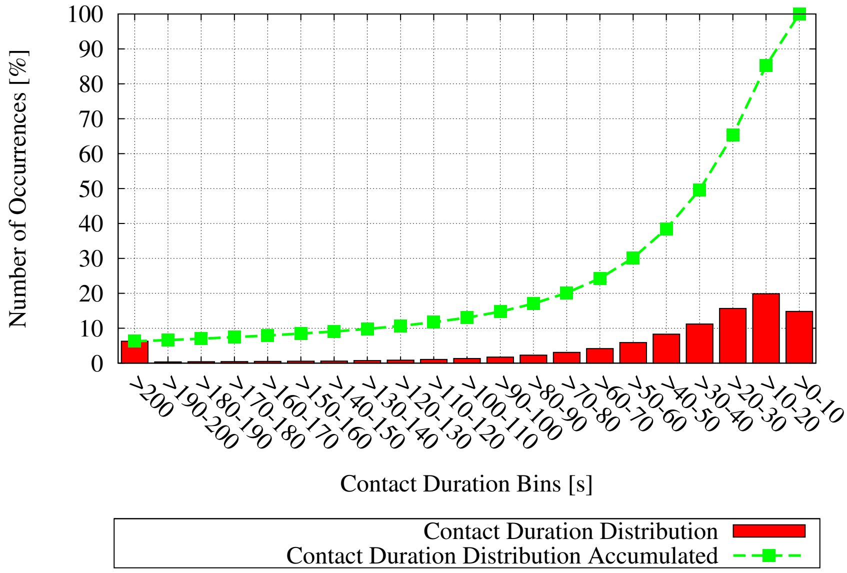

Figure 4.4: Histogram of vehicle contact durations for a range of 100m for all buses.

One specialty of public transport networks is that vehicles of the same line usually only have "encounter" and "following" contacts. We refer to the superset of these two sets as "overlapping". Lines are planned in a way that there is no intersection between vehicles of the same line. However, contacts between different lines can be each of the above types and depend on the predetermined routes and cannot be generalized.

4.1.1.2 Simulation with real Mobility Traces

The movement of vehicles in an urban environment is influenced by many external factors. Traffic lights, other vehicles, rush-hour congestion, breakdowns and construction sites are just a few examples. Because of this high complexity we decided against a synthetic mobility model for the simulations. Instead, we used our mobility trace of more than 1600 buses in Chicago. This trace was used as an external movement file for The ONE [74] simulator. The simulation scenario and the mobility trace are available for download[2]. Simulations are performed in 50m steps from 50m to 500m. This is not realistic for current 802.11a hardware, but these ranges are simulated since results can be applied to other wireless technologies and setups as well. For the simulation we chose a simple circular antenna radiation pattern. The influence of structures such as buildings and vehicles are excluded from the simulation. These two assumptions are reasonable since our work focuses on real world mobility in public transportation systems, and not on radio propagation and channel models.

[2] http://www.ibr.cs.tu-bs.de/users/mdoering/bustraces

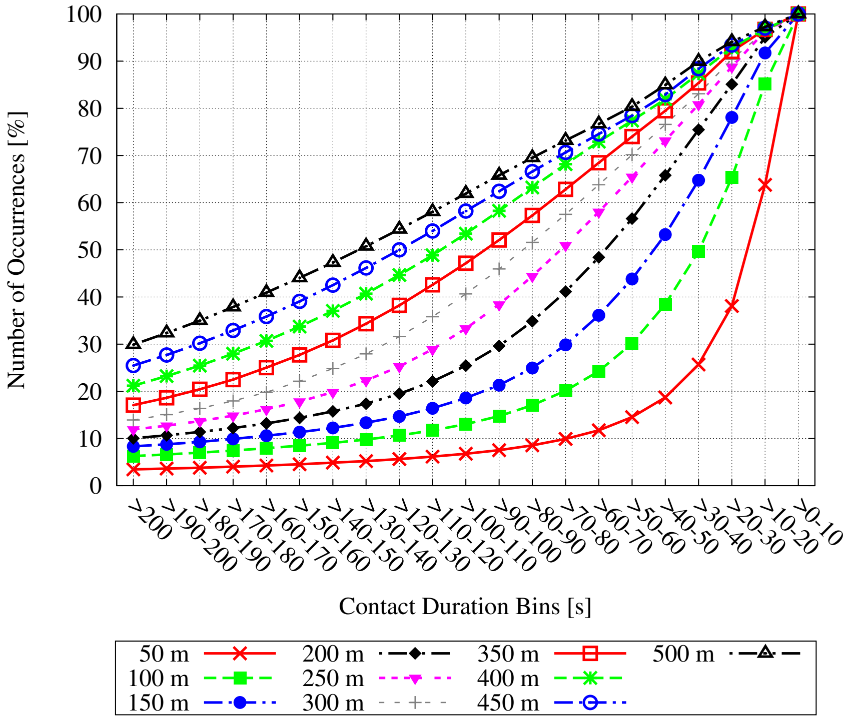

Figure 4.5: Accumulated probability distribution of vehicle contact duration for all ranges.

For the presentation of simulation results we start with an exemplary range of 100m (because this is an often used default value for WLAN simulations) and analyze all 161237 connections within the scenario. The bars in Figure 4.4 show the probability distribution of the contact durations. The statistical analysis results show that the median is 30s and 50% of the contacts are between 16s and 59s long. The graph above the bars shows the accumulated probabilities, i.e. the probability that a contact will last at least for a given time within the interval. This value is important for predicting the amount of data that can be transferred during a contact.

In figure 4.5 the accumulated probabilities for all ranges are shown. The graphs are interpreted as follows: A single graph represents a certain range. The x-axis bins show the duration of a contact. Together with the occurrences on the y-axis the diagram shows how long a certain percentage of contacts will last at least. For example, the 50 m graph shows that roughly 20% of all contacts last longer than 40 seconds.

As expected the average contact durations increase with higher ranges. However, it is notable that there is a steeper rise at low ranges, but generally on a lesser level.

In order to examine if the different types of contacts are affected differently by a variation of range, certain exemplary groups of vehicles were isolated. Figure 4.6 shows the median connection duration for these groups. First, the median of all vehicles in the scenario is plotted (denominated with “all contacts”). Then vehicles on two exemplary isolated lines are selected: lines 8 and line 21. This means that only overlapping contacts

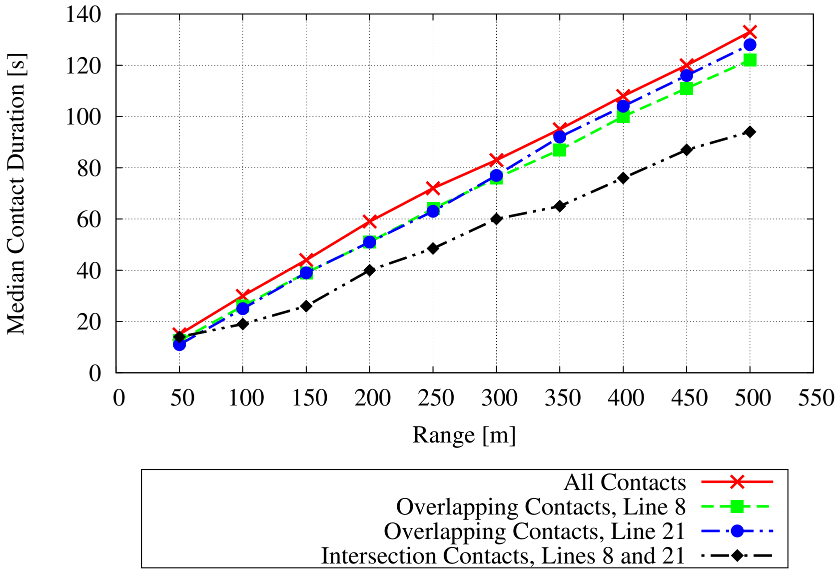

Figure 4.6: Median contact durations for different ranges for all buses.

are in these subsets. There is a slight offset between the medians of these two subsets and the set of all vehicles. This is due to average vehicle speeds on the selected lines, which is slightly different from the average speed of all vehicles. Nevertheless, these two graphs are still very similar to the graph of all vehicles. Starting at 350m, the graph shows that the median contact duration increases for line 21. This is caused by a higher number of vehicles and subsequently a lower distance between vehicles of that line. This leads to contacts between vehicles following each other in a certain distance on the same line. Intersection contacts between vehicles of different lines are not affected by this behavior, which is observable by the smaller gradient.

Figure 4.7 shows only contacts of vehicles on overlapping segments and figure 4.8 shows only intersection contacts. Contacts on overlapping segments have a duration median of 87s with 50% of the samples between 65s and 126s while contacts at intersections have a median of 65s and with 50% of the samples between 35s and 95s. Note that the distribution of contact durations also looks completely different in the case of intersection contacts. This shows that there is a higher variance because of external events such as traffic lights and traffic conditions.

This analysis of the two exemplary lines has shown that there are significant differences in the properties of certain classes of contacts. In the next section this behavior is investigated for classes of different contacts between all vehicles in the trace.

4.1.1.3 Classification by Contact-Angle

In the previous section basic types of contacts were isolated based on exemplary lines and routes of the operator's network map. Now we examine if the relative angle at which two vehicles make contact is an appropriate criterion for a more fine grained classification of

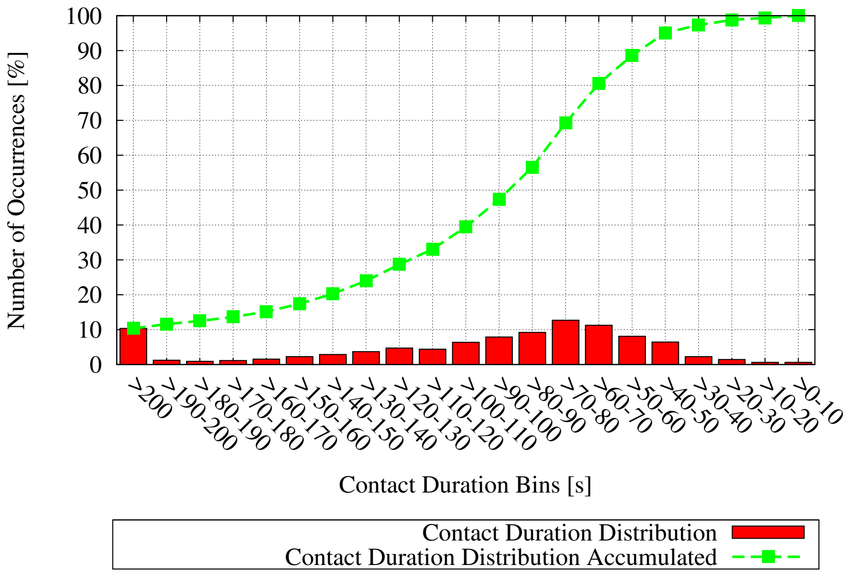

Figure 4.7: Histogram of vehicle contact durations for a range of 350m for overlapping contacts.

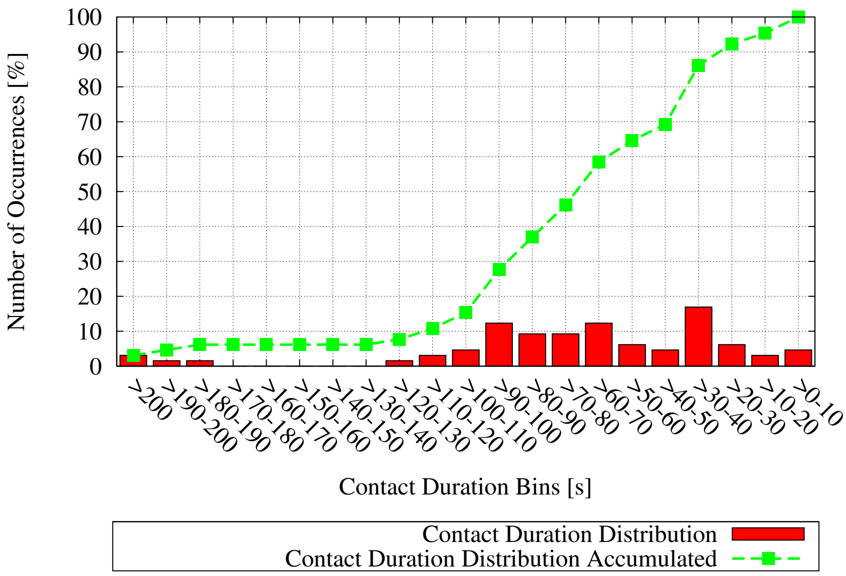

Figure 4.8: Histogram of vehicle contact durations for a range of 350m for intersection contacts.

contacts. The angle is determined by contact vectors, which start at the position of the vehicle at the beginning of the contact and end at the position at which the contact ends. We define the angle as the lesser angle between both contact vectors. Therefore the angle is between 0° and 180°. Using the angle is advantageous for practical reasons. In a real world public transport scenario the vehicles are usually equipped with GPS-receivers, so that the vehicle orientation is easily available. To calculate the relative angle of a contact the vehicles have to communicate each others orientation at the beginning of a contact. This does not require any additional protocol mechanisms since neighbor discovery (like IPND [86]) is already implemented and used to exchange information at the beginning of a contact. Routing and scheduling algorithms can exploit the relative angle to assign a contact to a class with certain properties. Therefore, it is possible to estimate contact duration based on the statistical properties of a class, resulting in better scheduling and routing decisions.

For the analysis we divide all contacts resulting from the simulation with the real mobility trace into five disjoint subsets. These subsets are based on the relative vehicle angle and named after the situation of the contact:

- **Encounter**: contacts with relative angles of $180° \pm 10°$ which means that vehicles drive in opposite directions. This is the most intuitive contact with two vehicles driving towards each other before making contact.
- **Following**: all contacts in which the vehicles relative angle is $0° \pm 10°$. This situation corresponds to a contact at an overlapping road segment in which both vehicles drive in roughly the same direction, e.g. when one vehicle follows the other.
- **Intersection**: Contacts which occur at an angle of $90° \pm 10°$, typically at intersections.
- **Not Moving**: contacts during which at least one vehicle is not moving. These are usually pausing or inactive vehicles. This means that it is not possible to calculate the angle since there is no movement vector for at least one vehicle. Nevertheless important information on the situation can be derived.
- **Ambiguous**: This class contains ambiguous contacts that cannot be assigned to a situation because the angle is not close enough to 0°, 90° or 180°.

A tolerance of $\pm 10°$ was allowed so that situations are also recognized with GPS inaccuracies, at slightly curved roads and at changing lanes.

4.1.1.4 Theoretical Analysis of Contact Duration

To gain a better understanding of the different situations in which contacts occur we start with a theoretical analysis of the contact duration. In this analysis, we make a number of simplifying assumptions. Namely, we assume that vehicles maintain a constant speed during the contact. We also assume that no obstacles block the communication of the vehicles and that the range has a circular shape around nodes. In addition, we assume that vehicles do not change the direction relative to each other during a contact.

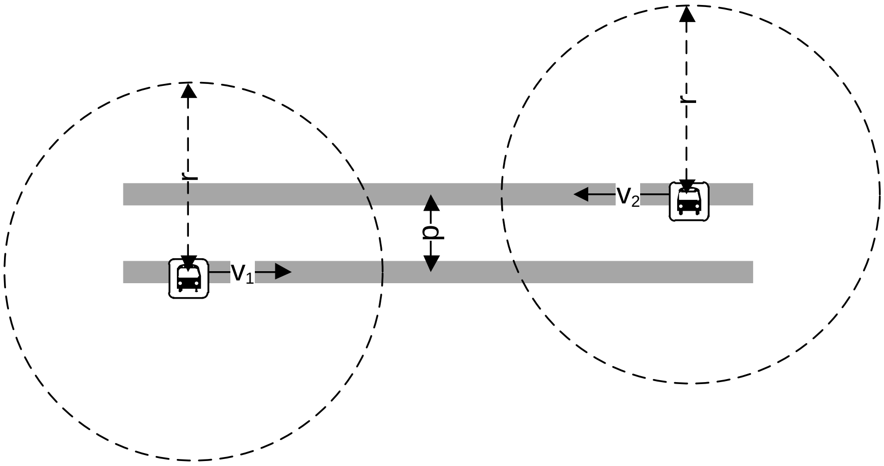

Figure 4.9: Encounter Contact in which two vehicles drive past each other.

Encounter

In an encounter situation as shown in Figure 4.9, two vehicles drive towards each other in opposite directions. As soon as the distance between the vehicles is below the range r, wireless communication can happen. After passing each other, the vehicles are still able to communicate while they are within range.

$$t_c = \frac{2\sqrt{r^2 - d^2}}{v_1 + v_2} \tag{4.1}$$

The contact time t_c depends on the speed of the two vehicles v_1 and v_2 as well as the range r and the distance between the vehicles while passing by each other d. Intuitively, this is the distance between the lanes of the road on which the two vehicles make contact. Subsequently, the time can be calculated according to equation 4.1.

Following

In a following situation as shown in Figure 4.10, two vehicles drive in the same direction but have a certain distance between each other. Depending on the relative speed, the two vehicles may overtake or stay behind each other. In practice, this contact type can be found when vehicles of the same line have a distance below the radio range.

$$t_c = \begin{cases} \frac{2r}{|v_1 - v_2|} & if\, v_1 \neq v_2 \\ \infty & else \end{cases} \tag{4.2}$$

The contact time depends on the speed of the two vehicles v_1 and v_2 as well as the range r and can be calculated according to equation 4.2. If the vehicles drive exactly the same speed, the contact would last indefinitely long.

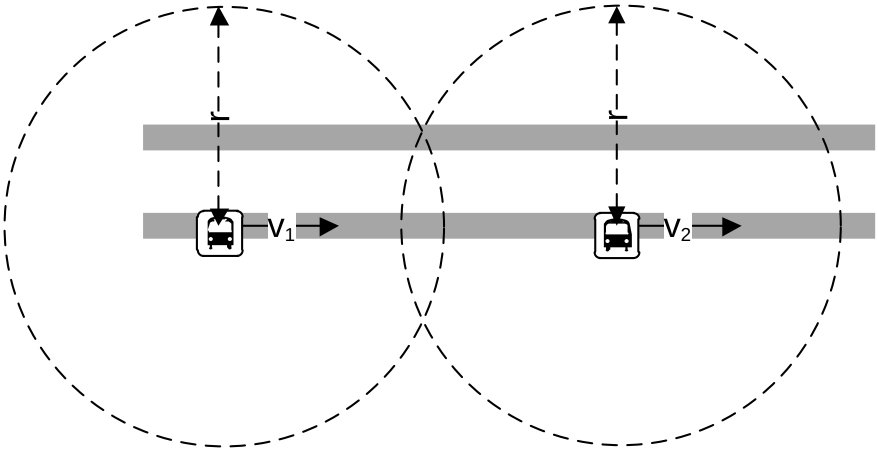

Figure 4.10: Following Contact in which two vehicles follow each other.

Intersection

Another contact type is two vehicles meeting at an intersection as shown in Figure 4.11. The vehicles can establish communication while approaching the intersection and stay connected while passing the intersection. The connection eventually breaks down as soon as the vehicles are out of range again. The direction of the two vehicles is orthogonal.

$$t_c = \frac{2\sqrt{2}r}{v_1 + v_2} \tag{4.3}$$

The contact time depends on the speed of the two vehicles v_1 and v_2 and on the range r. Equation 4.3 can be used to calculate the contact time under the simplifying assumptions that both vehicles are equally far apart from the intersection at the beginning of the contact.

$$t_{encounter} = \frac{2r}{v_1 + v_2} < t_{intersection} = \frac{2\sqrt{2}r}{v_1 + v_2} \tag{4.4}$$

Counter-intuitively, it can be seen, that without obstacles blocking communication, contacts of the type intersection are generally longer than contacts of type encounter. This is caused by the fact, that in an encounter the two speeds of the vehicles add up while in an intersection contact, the speeds are independent. Assuming a lane distance of $d = 0$, equation 4.4 shows the difference. Hence, it can be expected, that intersection contacts are $\sqrt{2}$ times as long as encounter contacts.

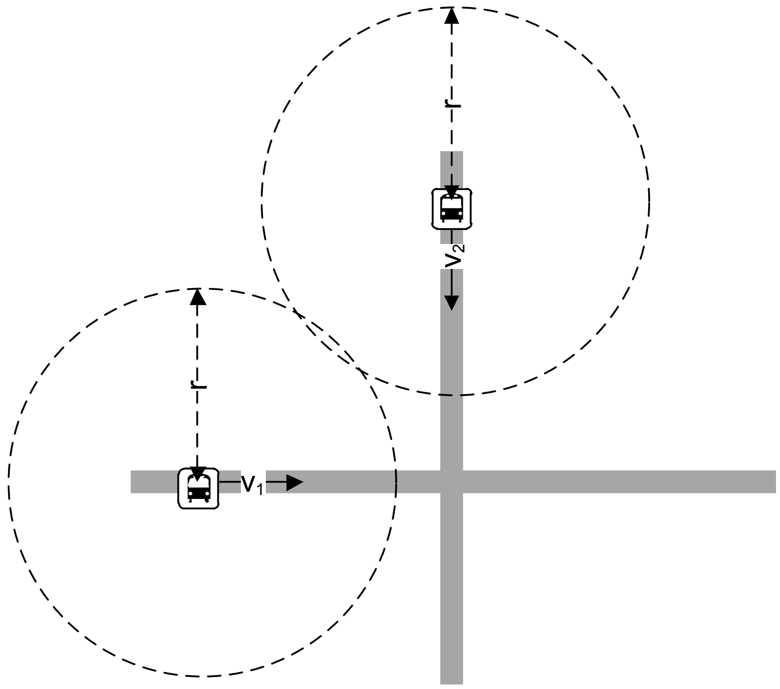

Figure 4.11: Intersection Contact in which two vehicles meet an an intersection.

Not Moving

A situation in which one of two vehicles that are making contact does not move is shown in figure 4.12. In this case, no contact angle can be calculated, since the not moving vehicle does not have a driving direction. In practice, this contact type can be found when one vehicle is waiting at a traffic light and the other vehicle is passing by. However, this case is similar to an encounter in which the speed of one vehicle is zero. The passing distance d is the minimum distance that the two vehicles have while passing each other.

$$t_c = \frac{2\sqrt{r^2 - d^2}}{v} \tag{4.5}$$

Hence, the contact time depends on the speed of the moving vehicle v and on the range r and can be calculated according to the equation above.

4.1.1.5 Characteristics of Classes determined by Contact-Angle

Now all contacts in our Chicago trace are classified by contact-angle. Figure 4.13 shows the number of contacts in each class. As expected, the amount increases at higher ranges. The class "encounter" contains more than twice as many contacts as "following" but shows a similar growth with increasing range. "Intersection" contacts are more effected by increased range. This can be explained by the likeliness of contacts, which is higher for

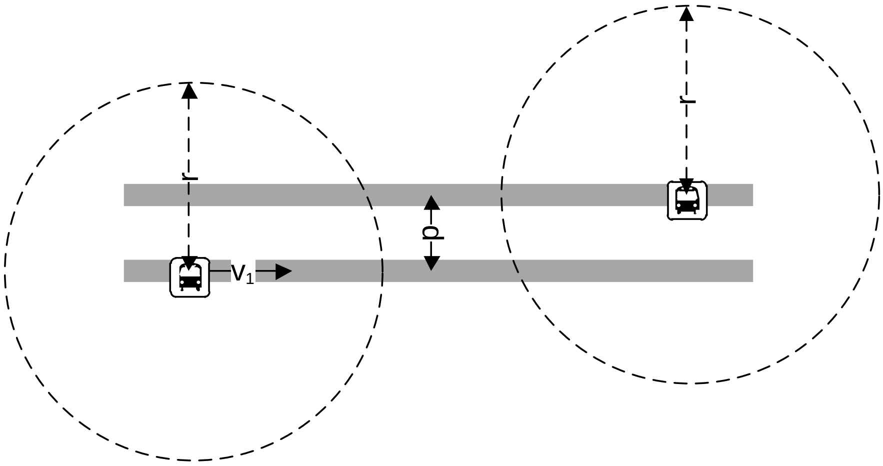

Figure 4.12: Not Moving Contact in which one vehicle drives past a stationary vehicle.

a larger range. Class “ambiguous” shows the highest sensitivity to range, because with higher range, vehicles can drive farther during a contact. This increases the probability that a vehicle changes its direction (e.g. by taking a turn or changing a lane) during the contact. Therefore it is more likely that the angle cannot be determined unambiguously. Nevertheless, the classes still show distinct characteristics, as described in the following.

The median duration of contacts is plotted in figure 4.14. At low ranges significant differences can be observed. Because of the higher relative vehicle speeds “Encounter” contacts are much shorter than “Following” contacts, which have the lowest relative vehicle speeds. With increasing range the differences between the classes get less distinct, but are still clearly visible even at 500m.

A quick glance at the cumulative distribution functions (CDFs) of the contact durations in figures 4.15 to 4.19 already reveals that there are special and clearly distinctive characteristics to each contact class. Each CDF “fingerprint” provides precious information to routing and scheduling algorithms, since it enables the prediction of contact duration. The CDF of class “encounter” in 4.15, for example, shows that a contact duration of 50s or longer for a range of 500m is very likely. Furthermore the CDFs are very useful for system design and capacity planning, for example when a certain performance is required and the antenna gain has to be dimensioned.

Lets take an exemplary closer look at the CDFs of “encounter” in figure 4.15 and “not moving” in figure 4.18. Apparently not moving vehicles are more likely to have significantly longer contact durations. The effect of increased range is also more pronounced and results in a gain of roughly 20s average duration per 50m additional range. For the range of 350m that we measured in section 3.1.2 this means that 90% of “encounter” contacts are 50s or longer, but 90% of the “not moving” contacts are 100s or longer. Based on our 802.11a measurements in Section 3.1.2, such contacts would allow a throughput of roughly 78MBytes (or more) per “encounter” respectively 155MByte (or more) per “not moving” contact in 9 out of 10 contacts. This example shows how big the influence of the contact situation on the performance can be. It also demonstrates the potential of the results for routing decisions.

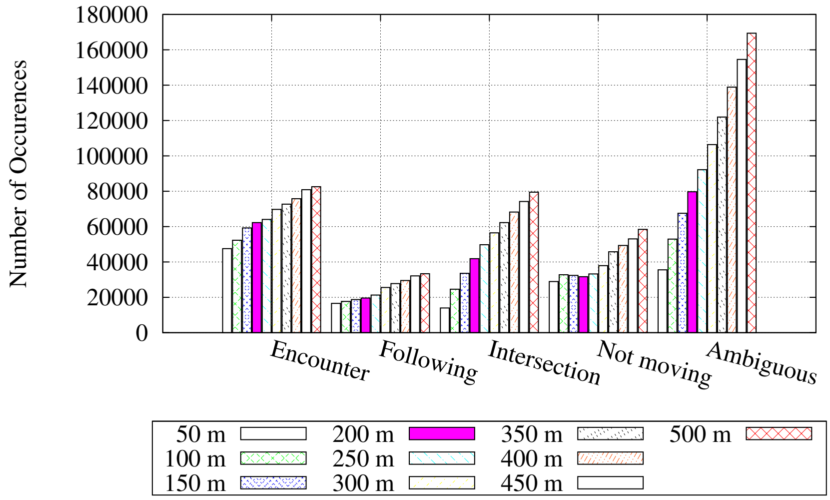

Figure 4.13: Absolute number of contacts in the five classes over ranges from 50m to 500m (50m intervals, from left to right).

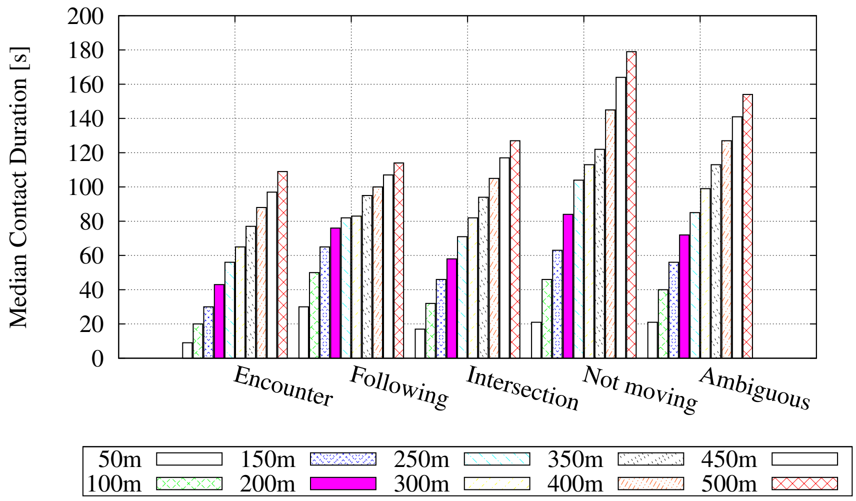

Figure 4.14: Median duration of contacts in the five classes over ranges from 50m to 500m (50m intervals, from left to right).

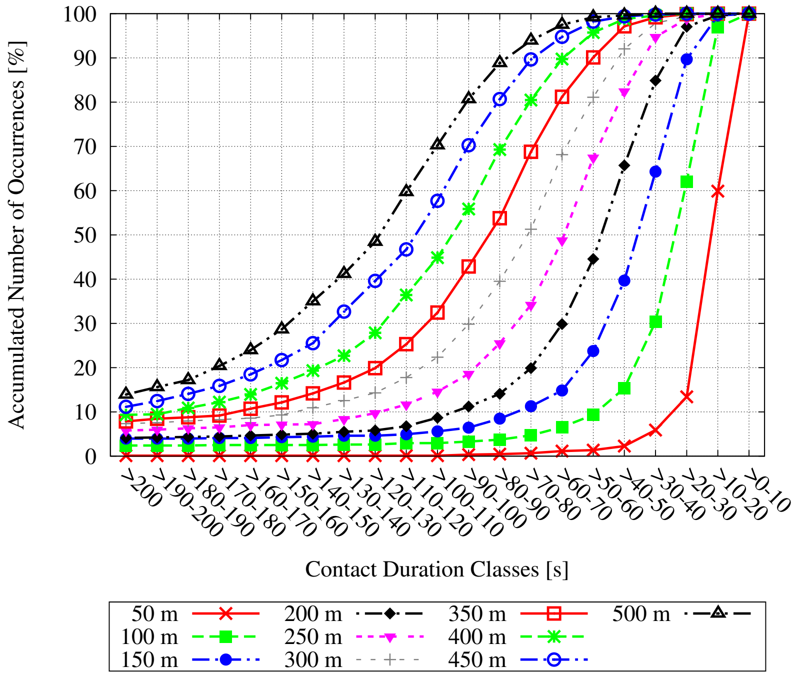

Figure 4.15: CDF of contact durations in class "encounter" over ranges from 50m to 500m.

Another interesting conclusion can be drawn from these results: as stated in equation 4.4, it can be expected, that “intersection” contacts are $\sqrt{2}$ times as long as “encounter” contacts. This is approximately confirmed by the results in figure 4.14. For a range of 100m, the median duration of “intersection” contacts is 32s while the median duration of “encounter” contacts is 20s. This is a ratio of 1.6. For a range of 200m, the ratio is 1.35 while $sqrt(2) = 1,414...$.

4.1.1.6 Verification of the Analytical Model

Looking at the expected growths of the contact duration according to our theoretical analysis from section 4.1.1.4 shows that in all described situations contact time should increase linearly with increasing range. To validate our analytical model, we have first calculated the relative growth of the median contact duration for each contact type over increasing ranges. Then, we have normalized this value according to the relative growth of the range. Intuitively, this means that a normalized relative growth of 1 resembles a growth of the median contact duration with the same ratio as the range has increased in one step. Figure 4.20 shows the result of our analysis. Especially “encounter” and “not moving” situations show a high variance in growth of median contact duration with changing ranges but still are within 10% of the expected value. “Intersection” and “ambiguous” contacts show the most stable dependency between range and contact

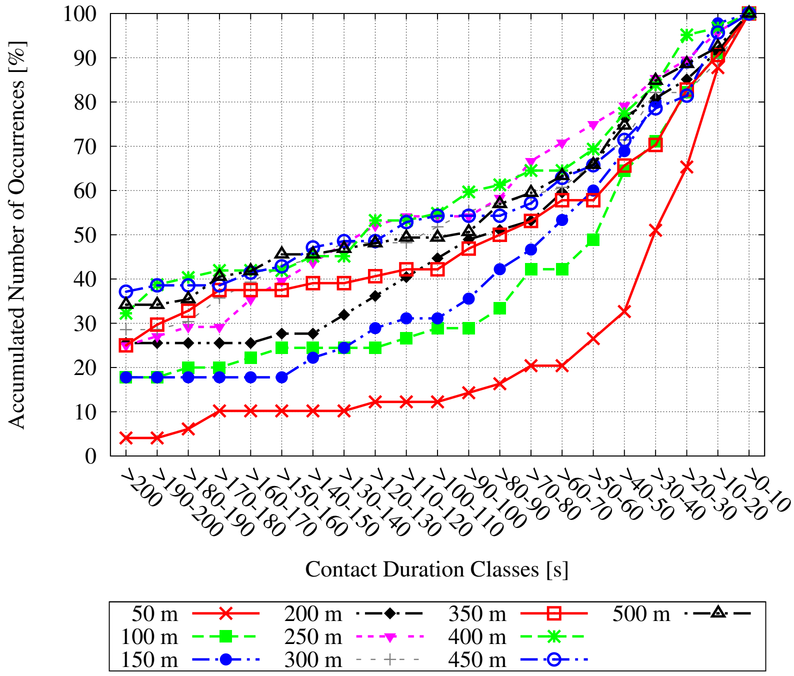

Figure 4.16: CDF of contact durations in class "following" over ranges from 50m to 500m.

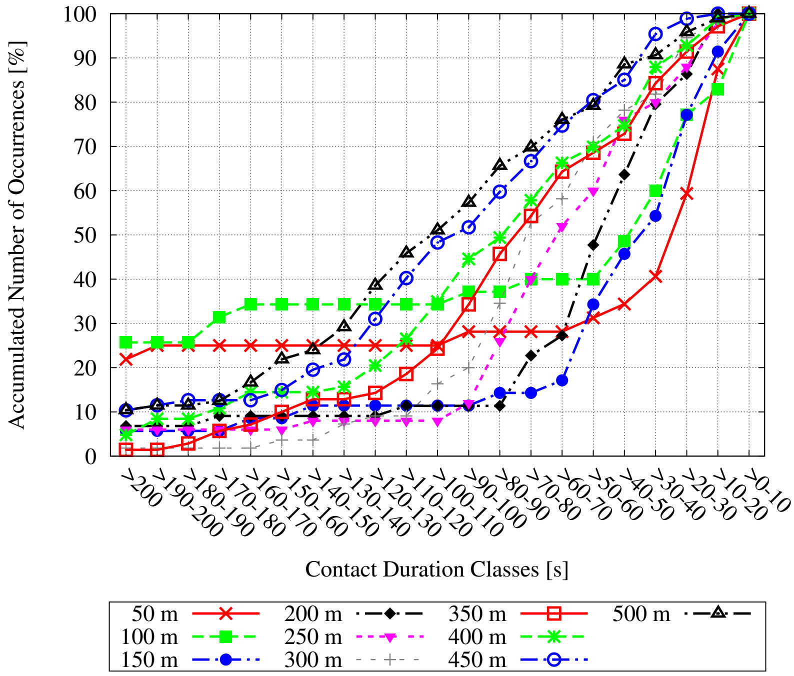

Figure 4.17: CDF of contact durations in class "intersection" over ranges from 50m to 500m.

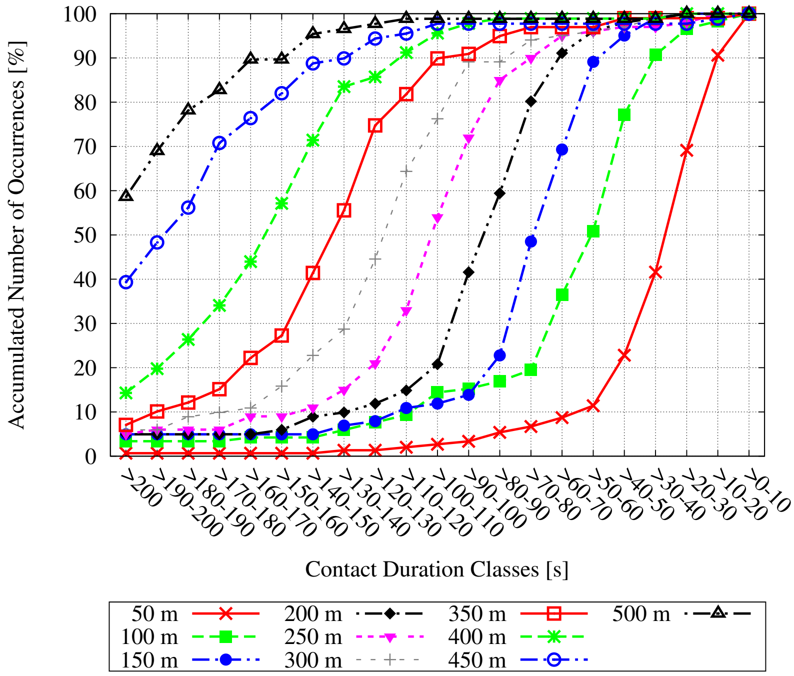

Figure 4.18: CDF of contact durations in class "not moving" over ranges from 50m to 500m.

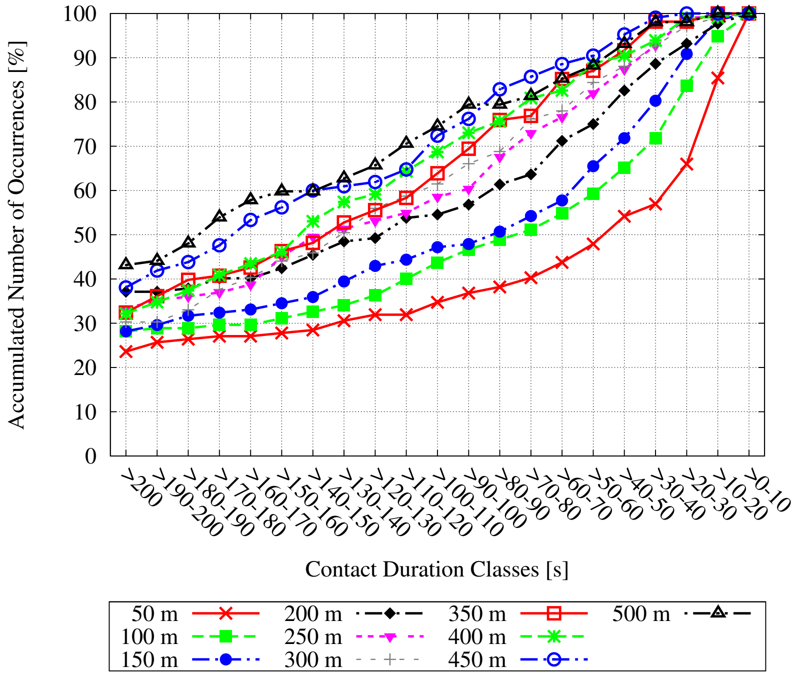

Figure 4.19: CDF of contact durations in class "ambiguous" over ranges from 50m to 500m.

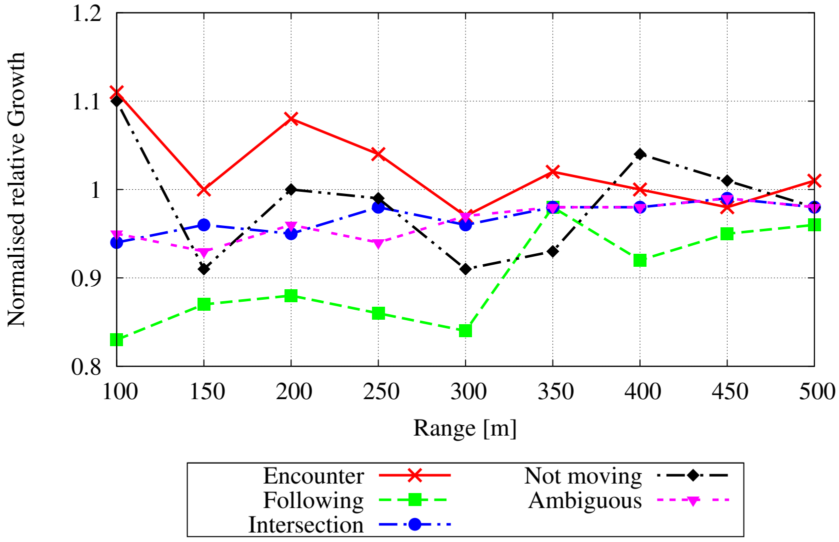

Figure 4.20: Growth of median contact durations over range normalized using growth in range.

duration, although it is slightly lower than expected. “Following” contacts depend less than linear on the range for ranges $\leq 300\ m$ and show almost expected behavior for ranges $> 300\ m$.

4.1.2 Conclusions

In this section an analysis of contact duration and contact characteristics in a simulation with real world mobility patterns of a large public transport network was presented. The statistical analysis of these contact times allows for a realistic estimation of the potential performance of a bus based DTN. It also provides important information on how dynamic the links in such a network are, and shows that some types of contact have a better quality than others. Knowing these properties is indispensable for designing and evaluating routing algorithms for these networks.

The simulation results provide important information on the expected performance of bus-based DTNs. We characterized different classes of contact situations and demonstrated how significantly different the probability distribution of contact durations in divergent contact situations is. The results and the accumulated probabilities of different contacts allow the prediction of contact duration, which is required for capacity planning and may also be helpful for routing decisions. We presented a simplified theoretical model for contact durations and have evaluated its applicability in practice. Moreover, we quantified the effect of radio range on contacts.

4.2 Bundle Transmission Scheduling

Due to the intrinsic properties of vehicular disruption tolerant networks, contacts between nodes exist only for a very limited amount of time. Therefore, for good performance it is important to transmit messages efficiently, and to minimize the waste of capacity by finishing the transmission before the contact ends, to lower the amount of data that is discarded or fragmented at the end of a contact. We investigate the performance improvements of various bundle transmission schedulers in simulations with real-world mobility. Moreover, we propose a new generic approach to scheduling. Although our approach has a positive effect, the evaluation also implies that the influence of transmission scheduling in realistic scenarios is lower than generally expected.

4.2.1 Importance of Scheduling in DTNs

Limited transmission time during a contact is often a bottleneck in vehicular disruption tolerant networks. Contacts only last for a short time, since both nodes usually are moving, and because the radio range is limited. Therefore, it is vital for a DTNs performance to use transmission time efficiently. A breaking contact causes incomplete bundle transmission. As a consequence, the bundle protocol agent on the receiving node either discards the incomplete bundle or stores a fragment. Unfortunately, reactive fragmentation is problematic with security (e.g. via bundle authentication blocks) and would cause significant overhead in replicating routing algorithms. Therefore, it is preferable to minimize the amount of discarded or fragmented data. A promising approach is to reorder the bundles in the transmission queue. Consider the simplified example in figure 4.21: A contact lasts for 10 seconds, and there are several small and one large bundles that should be transmitted and require 1s (small) and 7s (large) transmission time. It would be advantageous to send the largest bundle first instead of last, because this way the amount of discarded/fragmented data is lower. However, a real-world vehicular DTN is far more complex than this example. There are large amounts of bundles and nodes, contacts and contact durations (that are only statistically predictable at best), and possible interactions of bundle ordering with routing algorithms.

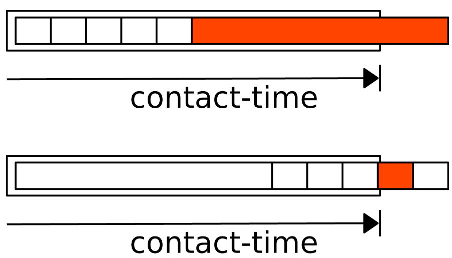

Figure 4.21: In this example the amount of discarded/fragmented data is reduced by rearranging the transmission queue.

To the best of our knowledge this bundle transmission scheduling has not yet been sufficiently researched. Most routing schemes use a FIFO approach, although there is previous work [87] on sorting the queue using routing specific metrics in order to increase delivery probability. Our work focuses on the evaluation of generic scheduling approaches that are independent of routing specific metrics, and therefore can be applied to various existing routing schemes. For this purpose we have implemented several different scheduling schemes for the The Opportunistic Network Environment[3][74], which we use along with our real-world vehicular mobility trace from chapter 3.2. In the evaluation the performance of schedulers is compared with the theoretical optimum and complete random scheduling as baseline. Simulations comprise several routing algorithms and a broad range of different configurations and scenarios. Moreover, based on the assumption that contact durations are predictable to a certain extent (as shown in the previous section), we design a scheduler that makes use of these properties.

The remainder of this section is structured as follows: First, steps during bundle transmission are discussed and optimization potentials are identified. Additionally, an overview of previous work in this area is given. Then various existing scheduling strategies are investigated and our own hybrid solution is introduced. Moreover, an approach regarding baseline and optimum schedulers for evaluation purposes is presented. Then the evaluation methodology and the simulation setup is explained. Furthermore, evaluation results of various existing scheduling and routing approaches are presented and compared to our own solution. In the conclusions we discuss the impact of our findings.

4.2.2 Bundle Transmission Basics

In a DTN with mobile nodes, contacts and the resulting links between nodes are usually available for a very limited amount of time. Therefore, it is important to use these resources as efficiently as possible. In the following enumeration the most important procedures during a contact are described:

1. Upon making contact, both nodes first have to identify each other to determine which bundles should be transmitted. This step is usually already included in the discovery protocol (e.g. IPND [86]), which is used to send out beacons that contain the node identifier.

2. Once the identity is known, both nodes independently select candidate bundles for transmission. This subset of all buffered bundles is made by the routing algorithm, which decides if the corresponding node is a suitable next hop. The routing decision is highly specific to the algorithm, but often based on the bundle destination.

3. Next, the nodes select bundles to be transmitted from the candidate bundles. The main point of this step is to prevent the needless transmission of bundles that are already buffered on both nodes, which is a very common situation in DTNs with replicating routing algorithms. Without communication between both nodes

[3] http://www.netlab.tkk.fi/tutkimus/dtn/theone/

(more exactly between the routing algorithm instances running on each node), it is not possible to determine which transmissions are unnecessary. Therefore, both nodes have to exchange a list of candidate bundle identifiers that are intended for transmission. More specifically, a common approach is to use 'smarter' data-structures instead of a list of identifiers, that use e.g. hashing for a heuristic identification. These algorithms are able to significantly reduce the amount of data that needs to be transferred in order to identify bundles for transmission. However, the drawback are occasional false positives, but on average the performance is still far better than with simple lists. Almost all replicating routing schemes use bloom-filters [88] for this purpose.

4. Now bundles are transmitted, usually in full-duplex mode (if the underlying convergence layers and transmission channels are duplex capable). In a mobile DTN transmission time is a very limited resource. The time needed to complete the previous steps reduces this time, which is the main bottleneck for the network performance.

5. Sooner or later, in a mobile DTN the connection breaks-down because of node mobility. Incomplete transmissions of bundles may result in bundle fragmentation (if this feature specified in rfc5050 [89] is supported by the bundle protocol implementation) or the incomplete bundle is simply discarded by the receiving node. Both cases result either in the waste of transmission time or in a more complex and resource hungry buffer usage, because fragments must be stored and reassembled, which causes a high overhead.

There are several starting points to optimize resource utilization during a contact. However, most of them have either severe drawbacks or only a marginal effect:

1. The discovery interval (also called beacon period in [86]) can be reduced to a minimum, but this also leads to a high overhead and increases interference. Since interference only matters if other nodes are in interference range, an aggressive discovery strategy (e.g. in the order of several milliseconds) may be acceptable in certain scenarios, especially if the discovery interval is dynamically adapted to network density. However, even in the worst case, only time $t_{di} + t_{tx}$ (with t_{di} discovery interval and t_{tx} discovery transmission delay, propagation delay is negligible in vehicular DTNs) is wasted. This means that the potential for optimization is limited, and that the absolute gain is decreasing with each reduction of the discovery interval.

2. The key to optimizing the selection of candidate bundles is to use efficient data structures and algorithms that prepare a selection before a contact happens. One approach to do so is to update candidates on bundle reception and to keep an index of possible contacts and transmission candidates for these contacts. This way, it is only required to lookup the candidates based on the corresponding node identity at the start of a contact, because all the processing was already done before. In fact, this is a “low-hanging fruit” approach that is easily implemented,

since most bundle protocol agents use common database engines with established high-performance indexing schemes for bundle storage.

3. Collection selection is a well researched area [90], especially in the context of content distribution in peer-to-peer networks, which is quite similar to bundle distribution in DTNs. Therefore, it can be assumed that the established algorithms are reasonably efficient and that there is not much potential for optimization in the summary vectors that are used for bundle selection.

4. Bundle transmission is the actual productive phase of a contact, therefore the goal of optimization is to minimize the time spend in all previous steps in order to have as much time as possible for bundle transmission. (Maximizing radio range and channel capacity also increases performance, but it is assumed that the best possible configuration is chosen during system design.) Since contact time is the main performance bottleneck in mobile DTNs, it is important to make the best use of the available time. Moreover, incomplete transmissions or fragmentation caused by a breaking contact should be reduced to a minimum. For example, it might be advantageous to prefer the transmission of several small bundles over a single large bundle, if it is predictable that the contact will end soon. This way, more bundles will be transmitted, so that the average latency is reduced. Moreover, if the contact breaks before the transmission is finished, less time is wasted (or smaller fragments are caused). On the other hand, this strategy may lead to "unfair" treatment of large bundles. Therefore, specifying and choosing a suitable scheduling strategy requires knowledge on the tradeoffs of each strategy. However, this problem has not been researched sufficiently before.

In summary, bundle transmission is the most promising candidate for optimization, because it is the longest phase of a contact. Therefore, scheduling is an important element of a routing algorithm, especially if different bundle priority classes are implemented as defined in rfc5050 [89]. For this reason the next section gives an overview of related work in the area of scheduling.

4.2.3 'Scheduling' in Related Work

Scheduling in DTNs has not yet received much attention. Some previous publications are available, but these are not consistent in the definition of scheduling. Therefore, this section gives an overview of the different focuses of scheduling related work.

In [91], the scheduling problem is formulated as "computing the contact times in which [...] data should be transmitted"[91]. It is assumed that contacts are "atomic"[91] and that "contact durations are negligible"[91]. Moreover, the proposed centralized algorithm is developed for DTNs in which "contact times are known (or can be accurately predicted) in advance"[91]. Moreover, the authors focus on forwarding but not on replicating routing algorithms.

The focus of [92] is on buffer management, i.e. strategies of dropping bundles if there is not enough buffer space available. Although this publications deals with a different problem, the optimization goal is the same - maximizing average delivery ratios and minimizing delivery delay. However, the authors show that these goals are competing by modeling the 'per message utility' for both goals. It is worth mentioning that this model assumes that contact durations are always long enough to exchange all bundles.

In [87], strategies for dropping bundles are termed 'queuing policies' and the order of bundle transmissions during a contact are termed 'forwarding strategies'. In contrast to [92], it is assumed that contact durations and bandwidth are limited, so that it is possible that not all bundles are transmitted during a contact. Therefore, the order of transmissions is recognized as an important parameter. The authors propose three forwarding strategies based on delivery predictability, which is specific to the PRoPHET [93] routing algorithm. Moreover, an additional generic strategy ("coin toss" random variable) allows for the comparison with other routing algorithms. Interestingly, the evaluation shows that 'GRTRSort' - a strategy that sorts messages by the improvement of delivery probability, and transmits bundles with the highest increase first - performs better than 'GRTRMax', which sorts by the highest absolute probability.

Yet another concept of message scheduling is presented in [94]. In the proposed ALARMS routing scheme messages from a region are buffered on the region's fixed gateway, and then transmitted to a ferry that travels to other regions. A region's gateway schedules bundles for transmission to the ferry. It is assumed that buffer capacity is unlimited, in contrast to bandwidth and contact time. The authors provide an evaluation that shows the positive effect of the proposed scheduling. However, the impact on practical applications is limited by the too rigid architecture in which nodes are either ferries, fixed region gateways or intra-region nodes.

In the following the mechanism that arranges bundles in a specific order for the transmission during a contact will be termed "bundle transmission scheduling". This work is focused on generic scheduling strategies, that are independent of any routing specific information.

4.2.4 Generic Bundle Transmission Scheduling Strategies

Generic scheduling strategies are independent of metrics that are specific to a certain routing algorithm. A bundle transmission scheduling strategy allocates transmission time to bundles for the duration of a contact. It is assumed that the contact duration is not deterministic and may end before all bundles are transmitted. A scheduling strategy is advantageous if it reduces bundle delivery delay or/and increases bundle delivery rates. For this purpose a maximum channel utilization during bundle transmission is required (i.e. no transmission time is wasted). It can be assumed that this basic requirement is fulfillable by any well implemented scheduling algorithm. Another starting point is to minimize the overhead that is caused by bundles that are not yet completely transmitted when the contact ends. An incomplete bundle is usually discarded, i.e. the transmission time is wasted. Some routing algorithms are able to deal with fragmentation, but this causes at least additional headers for each fragment and additional overhead if fragments are forwarded and replicated by other nodes.

There are several possible properties that can be used to establish an order of the bundles to be transmitted. A promising approach, for example, is to start a transmission with largest bundles first, and then continue with smaller bundles, so that only a small amount of time is wasted when the contact ends. This is based on the assumption that the probability that a contact ends increases after a certain time. Although this assumption is not necessarily valid for any possible DTN scenario, it is very reasonable for vehicular DTNs [69]. There are several other generic properties besides bundle size (e.g. bundle creation time, expiration time, current hop count, bundle priority, ...) that can be used to order the transmission buffer with ordinary sorting algorithms that have a worst case complexity of $\mathcal{O}(n \cdot log(n))$.

However, reordering bundle transmission buffers by certain properties leads to an 'unfair' treatment of bundles that are allocated to late transmission times. A potential drawback is that this might cause higher delay and lower delivery rates of bundles with certain properties if the network is near congestion. Because of the high complexity of vehicular DTNs it is very challenging to estimate the influence of scheduling strategies on performance. Therefore, determining the effect of transmission scheduling requires simulation with real mobility traces to gain realistic results.

In [87] it was shown that scheduling strategies that use utility functions which are specific to PRoPHET [93] may have a positive effect. Therefore, in this paper we focus on scheduling strategies that are more generic, i.e. we evaluate strategies that are independent of routing specific parameters. These have two advantages. First, they can easily be integrated to existing routers; and second, this allows evaluation across a range of different routing algorithms. In the following, various exemplary scheduling strategies are introduced.

4.2.4.1 FIFO/LIFO

These bundle scheduling strategies order the bundles in the transmission buffer by creation time, using the "creation timestamp" field in the bundle header as specified in rfc5050 [89]. FIFO means that the bundles are sorted by age and that the oldest bundle is transmitted first, then the second oldest and so on. This strategy is intuitively "fair" and widely used in DTNs. LIFO is the inverted strategy, it starts by transmitting the youngest bundle first. Therefore, newer bundles outpace older bundles. The expected positive effect is that bundles on shorter paths are delivered with a lower latency, resulting in a lower global buffer space utilization. However, the negative effect is that old bundles are not transmitted at all, if the network is near congestion, because newer bundles are always preferred. On the other hand, this will decrease the overall average latency and increase the overall number of successfully delivered bundles. This means that the positive global effect may outweight negative local effects. This theory will be investigated in the later evaluation.

4.2.4.2 BIG/SMALL

BIG and SMALL order bundles by their size. The basic idea behind these strategies is that an ending contact may have different effects depending on the size of the bundle that is in transit when the connection breaks. Because a large bundle requires more transmission time, it is more likely to result in an incomplete transmission. Moreover, this will either result in larger fragments or a larger amount of wasted transmission time because the incomplete bundle is discarded by the receiving node. Based on the fact that the duration of contacts in a vehicular DTN is not evenly distributed (as we have shown in the previous section), is can be assumed that reordering the transmission queue by bundle size will have an impact on the overall performance. Again, some bundles may not be transmitted at all, like described above in the LIFO strategy.

4.2.4.3 SWITCH

SWITCH is a hybrid strategy that combines BIG and SMALL. Again, the transmission queue is ordered by bundle size. At the beginning of a contact, this strategy behaves like BIG and starts with the largest bundle. After a certain time (at which the probability that the contact breaks increases disproportionally) the strategy switches to SMALL. Note that this change does not require resorting but just a reversal of the transmission queue. The expected result is a better utilization of transmission time but without discriminating large bundles.

However, it is important and non-trivial to choose an appropriate point of time to switch strategies. One approach is to use a link quality estimator (such as signal-to-noise ratio or bit error rate) and to assume that the contact will end soon after the link quality starts to decrease. The second approach is to predict link durations based on statistics of historical contacts. The feasibility of both approaches was evaluated with real-world measurements and the large-scale simulation presented in chapter 3.2. It is reasonable to assume that it is not required to make a 'perfect guess' of contact durations. Instead, we expect that 'being better than random' is sufficient for a positive effect on global performance, which will be validated in the evaluation. Although this approach requires at least some information on expected contact durations (e.g. statistics or context information on the type of contact as discussed in chapter chapter 3.2), it is still generic and applicable to any routing algorithm.

4.2.4.4 Benchmarks: RANDOM and PERFECT

To understand general limits such as lower and upper bounds, we introduce two further approaches as references. RANDOM is a benchmark scheduler that chaotically reorders the transmission queue based on a uniformly distributed random function. Therefore, it is a baseline scheduler used in the evaluation. Any scheduler which results in a DTN performance that does not exceed this baseline is not worthwhile, because it does not improve anything but still increases computing overhead.

PERFECT, on the other hand, is the upper bound of the positive effect that bundle transmission scheduling can have on a DTNs performance. It always makes globally optimal scheduling decisions, based on perfect knowledge (of both contact duration and the future routing path of the bundle). This assumption is 'perfectly unrealistic' in a real world DTN, but nevertheless an important metric in the evaluation. Unfortunately, there are several challenges in implementing this algorithm. First, it is not generic, since the upper performance bound can only be reached if the routing decisions in the DTN are also perfect. This is solved by a 'perfect' routing algorithm, which is based on Dijkstra. The second challenge it to gain perfect knowledge. Although this problem is - to the best of our knowledge - unsolved in the real world, it is not too challenging in simulation-based evaluation. In this context, perfect knowledge on future contacts and contact durations is gained by running the simulation twice. In the first run all contact events are recorded and used as input for the perfect router and perfect scheduler. Now the third challenge enters into the equation: Once the possible routing paths are identified by Dijkstra's algorithm, a perfect bundle transmission queue is calculated for each node. However, all buffers along each bundle's routing path are influenced, and each edge (contact) in the routing graph needs to be weighted with the remaining contact capacity. Moreover, a perfect scheduling fills the transmission time based on the knapsack problem. This means that the transmission time is the knapsack and the bundles the items. Calculating a subsum of all bundles that optionally fills the available transmission time is NP-hard, because the knapsack problem is NP-hard. Therefore, a perfect solution can only be calculated for a scenario with a limited number of bundles and nodes. Despite this limitation, PERFECT still provides very valuable information on the best possible solution (upper bound), which is an important metric to asses the quality of scheduling algorithms.

4.3 Scheduling Evaluation

In this section, the effect of the bundle transmission scheduling strategies that were developed and described in section 4.2 are evaluated. A significant impact would offer an opportunity to improve performance, since the mobility analysis in section 4.1 has shown that it is possible to make a basic prediction of the contact duration. Now, the question is if a smart scheduling strategy improves performance.

4.3.1 Evaluation Methodology

The purpose of the evaluation is to investigate if (and to which extend) the proposed scheduling strategies have an effect on the performance of exemplary real-world vehicular DTN scenarios. The common metrics are bundle delivery rate and bundle latency. However, for a fair comparison of scheduling strategies not only the number of delivered bundles must be taken into account, but also the amount of delivered data. Therefore, the sum of payloads of all delivered bundles is chosen again as a fair metric for the evaluation. This metric is global, i.e. it is summed up over all delivered bundles during a whole simulation run, and therefore describes the effects on the whole network. Besides a relative comparison of the performance metrics of different scheduling approaches, an absolute value of their quality can be given by using RANDOM (baseline) and PERFECT (optimum) as benchmarks.

Although it would be possible to use common mobility models or synthetic traces, we decided to use simulation parameters that are more realistic. For this purpose, we use the results of our previous work on real world vehicular measurements and mobility traces, and an in-depth analysis of contact durations in vehicular DTNs. In section 3.2.1.3 we reported how a mobility trace of Chicago buses was acquired, and gave an overview of the trace's properties. The analysis in section 4.1 showed that it is possible to clusterize contact durations, and that there are several types of contacts, with different properties resulting from different contact situations. Moreover, it is possible to statistically predict durations if the contact situation is known. We are drawing on these results for the evaluation of SWITCH, in order to investigate if this approach is advantageous in a real-word environment.

4.3.2 Simulation Setup

The basic setup is the same as in the previous section. It is also simulated with the real world Chicago trace. Again, 100 bundles with a random (uniform distribution) size between 10 and 100 MB are generated at the source. But this time the message buffers are limited to 2 GB, so that a possible effect of reordering at limited buffer space is observable. The simulation ends after the five hour time window of the mobility trace described above.

Switch Time The switch time of SWITCH is the parameter that triggers switching from BIG to SMALL. In a theoretical scenario this improves performance (as discussed above) if the timer is set to a value that is reasonably near the average contact duration.

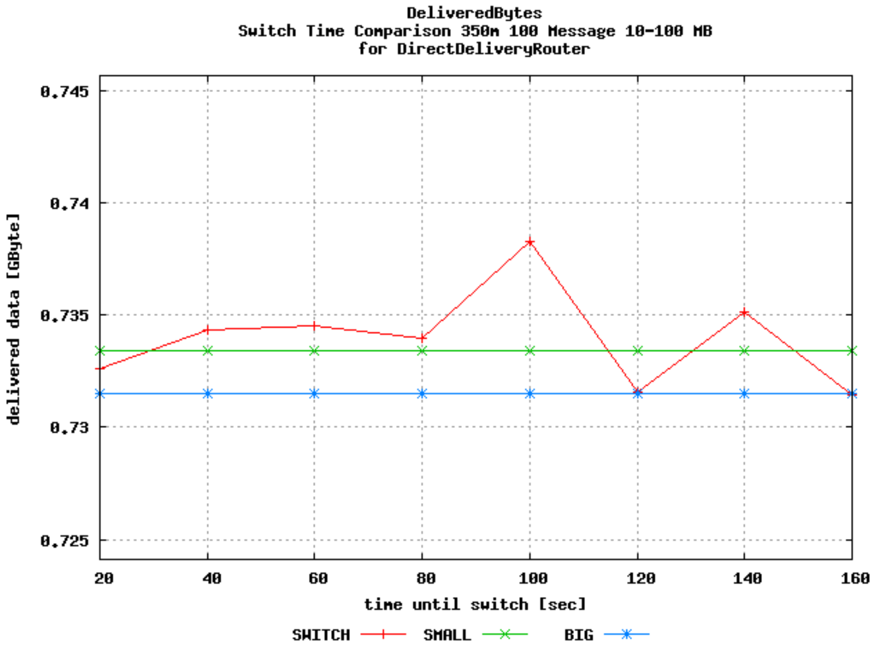

Figure 4.22: Performance of different switch times for direct delivery with 100 bundles.

However, it has to be investigated if this assumption is also true for real-world mobility. Therefore, the statistical distribution of contact times analyzed in chapter 3.2 are used to evaluate the effectiveness of SWITCH. This distribution is specific to the mobility trace and the radio range. In the simulation setup the median of all contact durations is 95s. Therefore, one would expect that the best performance of SWITCH is achieved with a switchtime near this value. For this reason times between 20 and 160 seconds were evaluated in steps of 20s. Again, the results are average values of 50 simulations with random source/sink-pairs.

Figures 4.22 and 4.23 show the performances of different switch times for direct delivery and perfect routing. These two were chosen as representative examples of a very simple and a very sophisticated routing approach. SMALL and BIG schedulers are given as a reference. As expected, the routing has a significant impact on performances. However, the point of this part of the evaluation is to analyze how different switch times impact the performance, therefore only the isolated values of a single routing are of interest. It can be observed that SWITCH performs best around 100s, which is near to the 95s median contact duration. However, it is also obvious that the effect is only marginal. This is not due to statistical variations, because results are averaged values of 50 different simulations. Nevertheless, the effect of switch time is far lower than expected.

Now, the same simulation setup is used but with a message size of 1-10 MB and 1000 messages. The results are shown in figures 4.24 and 4.25. Note that the total amount of data remains the same. With the smaller bundles the throughput increases. However, the effect of different scheduling approaches also changes. With smaller bundles the overall effect remains insignificantly, and moreover switch is no longer advantageous.

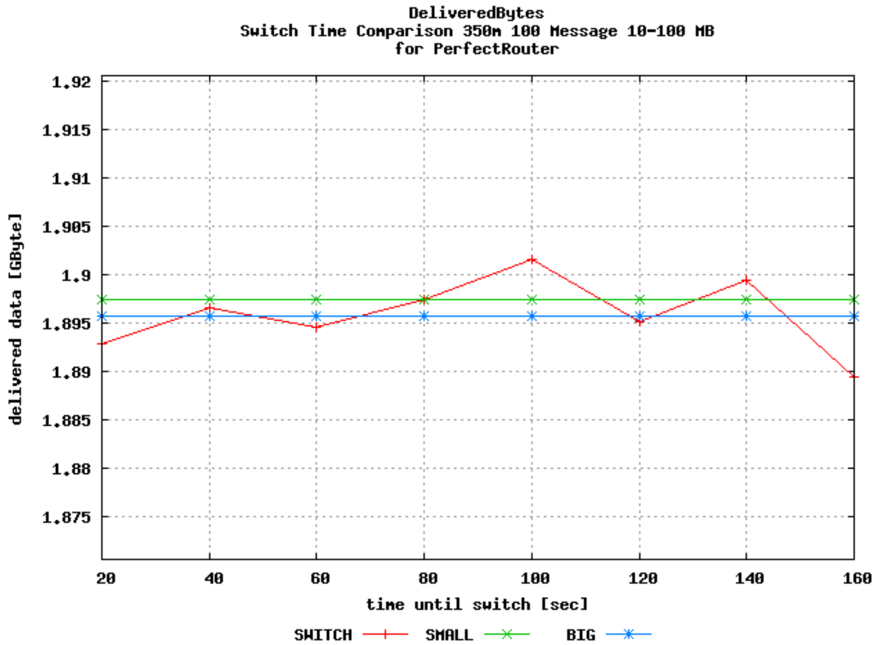

Figure 4.23: Performance of different switch times for perfect routing with 100 bundles.

There is a slight positive effect of BIG observable which is smaller than 1 %. As expected, RANDOM performs between BIG and SMALL. Moreover, it can be seen that SWITCH behaves exactly like SMALL at a switchtime of 0, which in fact means that it immediately operates like SMALL. Therefore it works clearly as intended.

The results with the Perfect Router in figure 4.25 basically show the same behavior, but again much better performance, which is simply due to the optimal routing. In figure 4.26 the upper bound of PERFECT scheduling can be observed. Even this combination with optimal routing and scheduling only gives a performance boost of roughly 5%.

4.3.3 Conclusion

We investigated starting points for the optimization of the bundle transfer phase in vehicular DTNs, and identified transmission scheduling as a promising field. Several scheduling approaches were implemented and simulated with various routing algorithms. We designed SWITCH, a hybrid scheduler, and evaluated it against several other approaches. Baseline and optimum solutions were used for comparison. The results show that the impact of scheduling is only marginal, and that the improvement by SWITCH is also not very significant, although a positive effect is observable. Since the simulation results show that even with a perfect scheduler (that is only theoretically possible in a real DTN) the performance gain would be marginal, we decided not to pursue the issue any further.

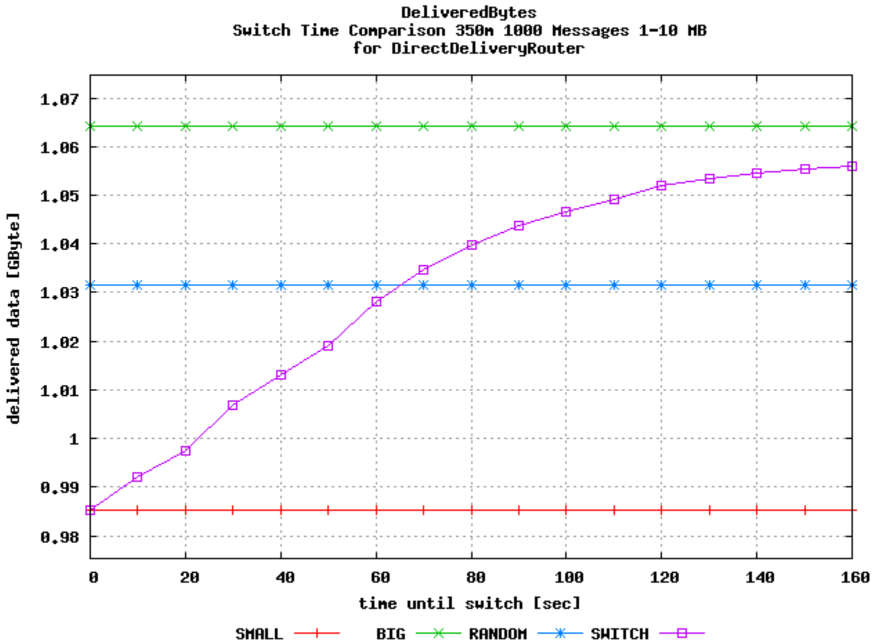

Figure 4.24: Performance of different switch times for direct delivery with 1000 bundles.

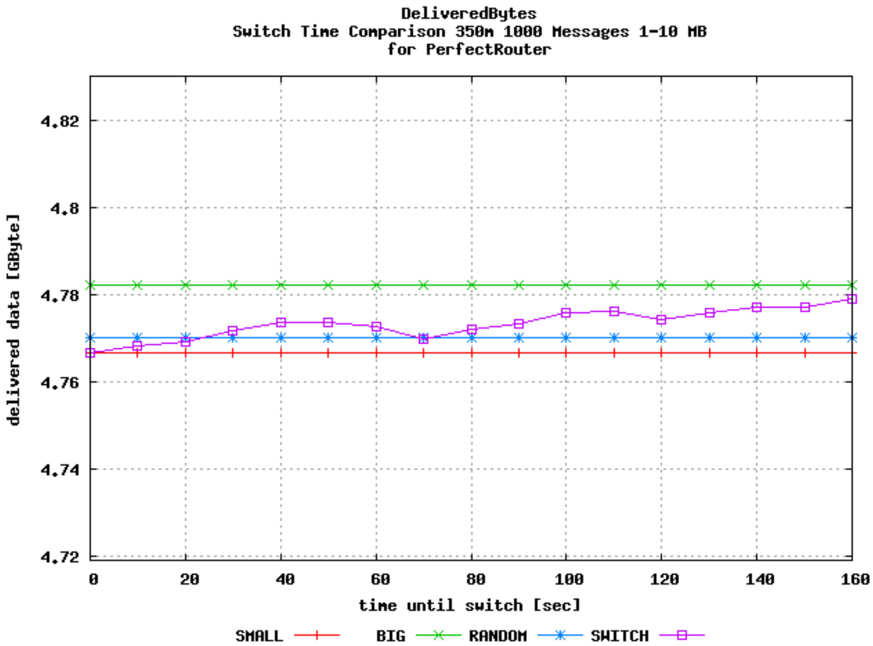

Figure 4.25: Performance of different switch times for perfect routing with 1000 bundles.

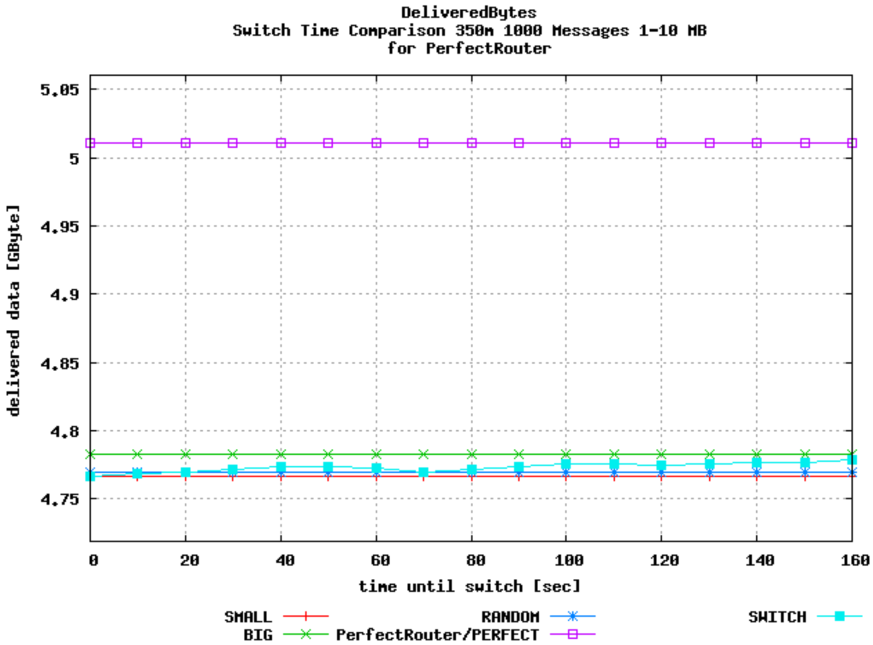

Figure 4.26: Performance of different switch times for perfect routing with 1000 bundles and upper bound of Perfect Scheduling.

Chapter 5

Routing in Public Transport

In the previous chapters the communication characteristics, mobility properties, and the different types of contacts between vehicles were investigated. Now, these results and the newly gained knowledge are used to develop a routing algorithm that is optimized for DTN in public transport. Moreover, this algorithm - and several previously proposed algorithms - are evaluated and compared in several simulations with our real-world mobility trace.

5.1 Requirements

The routing scheme to be designed is intended to make use of the specific properties of public transport networks. We aim at reducing resource usage and delivery delays by designing the router less generic. This means that the router will only be useful in a public transport scenario. It could be argued that such an approach would create some kind of proprietary DTN that only public transport vehicles can connect to. However, there is no specification that prevents a DTN node from using several different routing protocols. Therefore, it is an option to use a specialized protocol within a group of nodes, and a more generic protocol with external nodes. The big advantage of a specialized protocol is the usually very application specific deployment area of a certain DTN, that allows the exploitation properties that these application specific nodes have in common. In the public transport case these are the well defined vehicle routes and timetables, that make node trajectories well predictable.

As prerequisites it is assumed that nodes are aware of their location and that timetables and routes are available in a machine readable format. The location accuracy is assumed to be at least good enough to determine that a vehicle is on a specific route segment between two stops. This low accuracy ensures compatibility with older signpost/odometry based systems. However, a higher accuracy is generally advantageous, as it will result in a better temporal resolution of routing decisions and therefore in a potentially better performance.

Based on our analysis of communication characteristics, public transport mobility, and contact properties, we identified the following requirements for a DTN routing scheme in public transport networks:

- Routing decisions are based on a high-level description of the public transport network. The routing must be able to predict future contact opportunities between vehicles. Moreover the router must build source-to-destination paths based on these contacts.

- The mobility analysis in chapter 3.2 has shown that there is a relatively high deviation from the timetables during rush-hours. This will regularly cause some of the predicted contacts to fail. Additionally, unpredicted contacts may occur. The routing algorithm must react to failed contacts and recalculate a suitable new route. Because of the DTN paradigm this has to happen on the intermediate node with the failed contact, not on the source node. Moreover, if timetable deviations lead to unexpected additional contacts, the algorithm has to check if this new opportunity allows for a better route.

- In the contact analysis in section 4.1 different types of contacts with different distributions of contact durations were identified. Especially intersection contacts and contacts on overlapping route segments have different probabilities for being successful. In cases where the type of contact is already predictable based on the lines that the vehicles are operating on, the routing algorithm should weight the expected quality of contacts. Since a route usually comprises several contacts, this allows for the calculation of a "route success" indicator. In result, several route options can be compared not only based on predicted delivery delay but also on the probability of successful delivery without route recalculation.

- The next requirement is a low resource utilization. Extensive replication like in Epidemic [18] causes significant overhead. On the other hand, limited replication, like in SprayAndWait [36], makes routing much more robust. Therefore, a very controlled and well limited replication strategy is required in order to get some robustness against failed contacts. The design goal is to keep the message buffer utilization low by implementing a more restricted and more targeted replication strategy. Thereby, the total number of replicates in the network is reduced, which has a positive effect not only on buffer space but also on resources such as channel utilization (meaning a more efficient contact time usage) and processing time on the nodes (because there are fewer routing decisions and message database operations).

- As a final requirement, the router has to be desired to be suitable for real world applications. This means that it must not require any manual interaction or rely on assumptions that are not practical in real world public transport systems. Examples for such assumptions are to imply that vehicles are dispatched to lines following defined rules, or that a certain vehicle operates statically on a certain line. The trace analysis in section 3.2 has shown that this is not always the case in reality. Moreover, during our cooperation with a public transport supplier (BBR Verkehrstechnik GmbH, Germany) and a public transport operator (Braunschweiger Verkehrs AG, Braunschweig) we learned that most dispatching decisions are either first-come-first-serve (e.g. the first driver arriving at the beginning of a shift grabs an arbitrary vehicle at will, and is dispatched to the first line in the schedule), or in the case of disturbances simply situation-specific "gut decisions" based on the experience and intuition of the dispatcher.

5.2 Data Models

Data models of DTNs are more complex than models of 'traditional' ad-hoc networks which only take end-to-end connectivity into account. The reason for this is that in DTNs routing paths may also include nodes carrying messages from one location to another, and not only transmissions over links that change between active and inactive over time. Therefore, modeling a DTN requires an additional dimension, as the network changes over time and space. Such a model - a *space-time graph* - was first proposed in [95]. More specifically, the authors define a space-time routing framework for networks with predictable movements. These movements are either finite over time T or infinite because of periodic movement of the nodes. The basic idea is to generate a routing table which is in fact a two-dimensional matrix (destination and time), that is used to calculate the sequence of local forwarding decisions on each node.

Instead of listing all possible paths and searching for the optimal path on demand, which is a very expensive operation (because of the potentially exponential number of paths), the authors propose a space-time graph model and an algorithm to compute optimal paths efficiently. From these graphs the routing information for forwarding decisions is obtained. One of the key simplifications that reduce complexity is that the model is defined with discrete time intervals. This allows for a "*layered space-time graph*", in which layers represent snapshots of the network at discrete time intervals. Usually, "*time-copies*" of a node are vertically connected across these stacked layers (therefore forming columns) to represent message-carrying. Message transmission is represented by directed edges between columns, that skip layers (time intervals) to represent transmission delays. Now, it is possible to implement shortest path algorithms such as Dijkstra or Floyd-Warshall to get the lowest delay route on the resulting graph. However, both require adaptations, because the time component is not taken into consideration by the original algorithms.

A further approach to reduce the complexity of a DTN routing graph is described in [33]. It aims at reducing resource consumption by creating a network model with a sparser structure. However, the authors point out that classical topology control mechanisms used in MANETs or WSNs cannot be applied. These mechanisms are based on constructing geometrical structures or creating hierarchically organized clusters, but do not take the two-dimensional space-time properties of DTNs into account (in other words: they do not respect non-end-to-end paths that result from node mobility). The authors show that respecting this property increases complexity of this already NP-hard problem. Therefore, they propose heuristics based on greedy algorithms and show the effectiveness of these heuristics by simulation.

The basic graph model [95] and the topology control heuristics [33] are both applicable to the development of a routing algorithm that is specialized on public transport networks. One might argue that the discretization of time is too unrealistic, although it allows for a great reduction of complexity. However, the behavior of a DTN based on urban traffic vehicles can only be predicted with a significant uncertainty, since there are always variations in the movement because of the influence of traffic conditions. Therefore, from a practical point of view, the imperfection of the model is very acceptable. Another point regarding the uncertainty of future node movement is the time horizon. In theory, a public transport network has a periodic mobility pattern which is defined by the timetables and lines. But in practice vehicles are often swapped or change lines (as described earlier), and dynamic changes are made due to changing demands and due to road or traffic conditions. Therefore, it is neither practical nor required to have an infinite time horizon. In fact, most public transport vehicles return to a central garage or depot at the end of a shift. This means that a time horizon of a few hours is sufficient for most applications.

5.3 Design

In this section we present our design of a DTN routing algorithm for public transport systems. Our approach aims at a specialized algorithm that actively exploits the characteristics of public transport systems to achieve a better efficiency. The drawback of this approach is that the specialized routing cannot be applied to such a wide range of applications as more generic approaches like epidemic routing. On the other hand, DTNs are typically deployed in very specific scenarios, so a potential performance gain is often more important than generic routing.

5.3.1 Approach

The aims in the design of Routing in Urban Public Transport Systems (RUTS [96]) are resource-friendliness as well as timely and reliable delivery as far as possible in a public transport vehicular DTN. In addition, the modular design allows for changes and extensions in the future.

Urban public transport has some specific characteristics, e.g. timetables and network maps that can be exploited for routing. If this context information is available on each DTN node, an efficient routing is possible. First the possible contacts between the vehicles are determined and stored in a routing-graph. Based on this routing-graph, different ways from the sender to the receiver can be identified. Each way represents a possible routing path for a bundle. Due to external disturbances and local conditions, contacts have different performances and probabilities. Therefore, every involved contact in the routing paths has to be evaluated. This result is a rating for every possible routing path. Now, the bundle can be passed on the most suitable path. Multiple bundle copies can be used to increase the delivery probability and to reduce latency. After the routing path for a bundle has been determined, it is stored in the bundle. Now, the routing path can be followed unless there is a discrepancy which occurs when a planned contact fails. Only in this case a new path has to be calculated.

5.3.2 Basic Routing Sequence

At first, the number of bundle copies has to be determined. Copies can increase the delivery speed and rate, because as a result of unpredicted disturbances it may happen that not every bundle can be routed on its calculated path. We define n as the number of bundle copies (including the original bundle). n is usually small, and $n \geq 1$. Consequently, $n = 1$ if no bundle copies are used. For each new bundle the transmitting node sets the initial value of n into an additional field in the bundle to save storage and transmission capacity for 'real' copies. As long as $n > 1$, the bundle may be split by a DTN node to follow a previously calculated path for each bundle copy. This means that a bundle is transmitted with an adjusted value of n to another node whereby multiple transmissions can be saved. Now, both nodes have the same bundle and the sum of available copies of both parts corresponds to the value before. Over the whole bundle lifetime, it must be ensured that the number of copies in the DTN network is always $\leq n$. In addition to the splitting, it is still possible to forward bundles to follow the calculated paths.

In order to send the bundles and their copies on different routing paths, possible ways have to be detected. Therefore, k routing paths have to be identified where typically $k > n$, in order to get a higher choice because different routing paths can have a different quality. It is possible that $k \leq n$, if not enough paths exist. k has a direct impact on the computation resources that are used, and therefore is parametrized during deployment on a node with specific hardware capabilities. To find the necessary paths RUTS uses the available timetables and the network map from the urban public transport system. This context information can be used to calculate possible contacts between vehicles. Each possible contact is stored in a special routing-graph. In the first step the search starts from the sender DTN node. All possible contacts are inserted into the routing-graph. Now, the search will be resumed for all nodes contained in the routing-graph. The search is iteratively continued and it is stopped if at least k potential paths are available. In section 5.3.3.1 the graph is explained in more detail.

Now, the k fastest paths from the transmitter to the destination must be found in the routing-graph. As already mentioned the paths vary in quality such as different contact probability and duration between the vehicles. Therefore, the paths are evaluated and the n best ways are selected for forwarding. In addition a score value for each of the k paths is calculated. It reflects the estimated time required for the transport from the sender to the receiver, the number of required hops and the line characteristics for each DTN node. All these data can be obtained from the routing-graph and the available context information. For the line characteristics the driving directions and the kind of contact is crucial. In section 5.3.4, this issue is addressed in detail.

The score for each k-path is determined as follows: TT indicates the calculated transmission time from the transmitter to the destination, RPT defines the rating for a possible transmission and NT declares the number of bundle transmissions to reach the destination.

$$Score_k = TT + \frac{\sum(RPT)}{NT}$$

Each parameter can be parametrized to control its impact on the score. In addition, the rating-models for the possible DTN node transmissions are interchangeable. After each of the k paths has been rated, the bundles are sorted according to the scores and distributed on the most suitable paths.

5.3.3 Routing-Graph

5.3.3.1 Construction

As described above, the routing-graph is used to determine possible routing paths. Therefore, it is a data-structure that stores potential DTN routes between source and sink nodes. Figure 5.1 shows an example for such routing-graphs and the resulting paths whereas the nodes $A, B, ..., H$ symbolize specific DTN nodes for any scenario. In this example, a bundle is transmitted from node A to node H. The directed edges represent expected contacts between the vehicles, so that a contact, e.g. between the nodes A and C, is possible. These contacts result from the analysis of the context information by comparing the positions and schedule times of the vehicles. As a special feature, the routing-graph stores the time when a node has been inserted into the graph, except for the destination node. In the example, the creation time of each node can be seen in the timeline, and it always starts with $t = 0$. Typically, the times represent minutes because the timetables are usually defined in minutes, too. This additional knowledge is needed to determine the weights from the edges. Here, the weights are the difference between the times of the timeline from two corresponding nodes. These represent the time which will probably elapse before the next appropriate contact happens. Later, the predicted time to transfer a bundle can be read in the sum of the weights from a routing path.

Once a possible new contact has been found for any node in the graph, the respective node is inserted into the graph. The position in the graph is dependent on the current considered time step. In addition, a directed edge must be inserted, from the node which found the possible contact to the newly added node. The weight results from the currently considered time step and the time step from the associate member node:

$$Edge_{Weight} = T_{Current} - T_{MemberNode}$$

5.3.3.2 Splitting

Over the whole time it must be ensured that the graph is cycle-free to avoid invalid ambiguities. This requires a cycle test after inserting a new edge. If necessary, the last inserted contact must be removed. However, nodes may have multiple incoming edges. This appears if more than one possible contact to a node exist. Therefore, these nodes are split into a representation that prevents ambiguities by a unique assignment. For example, in figure 5.1, the former node F has been split into F_1 and F_2, because the nodes B and D had a possible contact to former node F. If a node is not split this results in invalid paths in the graph, because a differentiation between possible contacts at different times is no longer possible. Therefore, nodes must be split as soon as a second incoming edge is inserted. In the example the node F has a first contact at $t = 2$ with node B and a further at $t = 4$ with node D. After a node has been split, the further search for new contacts is performed only once. Therefore, not every single subnode must be searched individually. If new contacts are added, edges from each sub node towards the contact node are inserted. Multiple contacts between the same nodes at different times are possible. In such cases, only the first contact is considered because

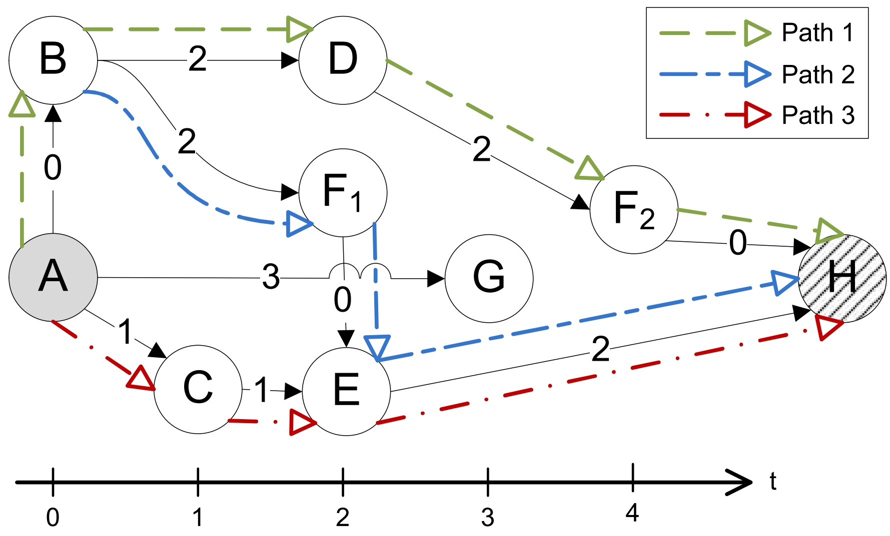

Figure 5.1: Example for a RUTS Routing-Graph.

no additional benefits can be drawn from later ones. Normally, the search stops at k incoming edges of the destination node.

5.3.3.3 Routing-Graph Usage Example

At the beginning in figure 5.1, the graph contains only the source (A) and the destination (H) node. The source node is located at $t = 0$ on the timeline, because the search starts always with this time step, which represents the current time. The destination node is outside the timeline because it can not be assigned a fixed value, because different routing options result in different times.

Based on the timetables and the network map, new possible contacts are iteratively searched for each time step and all nodes in the graph, except for the destination node. The search is terminated as soon as the destination node has at least k incoming edges. In the illustrated example, at least two potential paths should be sought ($k = 2$). For illustration the example shows 3 paths.

Moreover, gateways and relays can also be included in the routing-graph. Gateways are stationary nodes and allow transmissions to the backend system over a WAN. Thus, multiple backends can communicate directly with each other. Relays are also stationary but have no WAN connection. Through the usage of relays and gateways the delivery probability and reactions to external failures such as breakdowns and delays are improved. Once a node has a possible contact to any gateway, all gateways are inserted into the routing-graph as nodes with a minimum weight for the edge because an immediate transfer is possible. Relays are added individually because they have no direct connection among each other. Thus, the relays and gateways are involved in the further search.

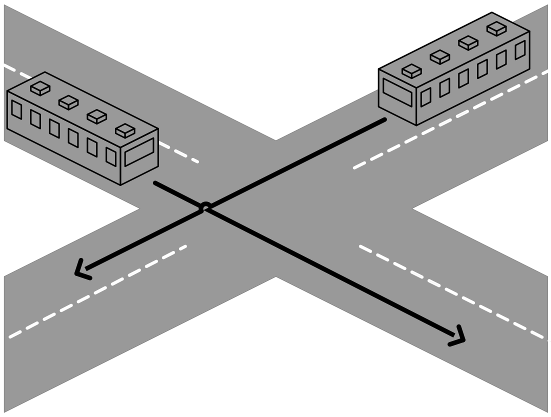

Figure 5.2: Bundle transmission at an intersection.

Finally, the current delay from the transmitter node can be included in the search for possible contacts. This can be deduced from the position of the vehicle and the timetables. Therefore, the identified delays must be considered while looking for new possible contact options by adjusting the own timetables with the delay.

5.3.4 Weighting of a bundle transmission Opportunity

As already described in section 5.3.2, the probabilities of all possible contacts between vehicles must be identified and weighted. This is an indicator for the quality (or confidence) of a route, which we denominate “score”. Currently, RUTS uses a simple weighting model and differentiates between two use cases. The first case is shown in figure 5.2, the intersection of two lines. If there is no additional dynamic context information available such as the duration of previous contacts, a low contact probability has to be assumed. Thus, the weight for this possible bundle transmission is defined as 1, which can be parametrized with an additional factor to regulate the impact on the route score.

Overlapping segments of the lines are much more complex. In this case lines overlap on a section of the network map. The length of the overlapping segments and the direction of the vehicles are of relevance. Figure 5.3 shows an example with vehicles which drive towards and after the contact they drive away from each other. The other possibility is that vehicles drive in the same direction. Both possibilities have a different contact probability and expected contact duration. The direction of the vehicles can be derived from the timetables and the network map. In both scenarios the weight for the score is determined by the length of the overlap. The length equals the number of edges in the routing-graph which cover the overlap.

The presented weighting model allows a simple differentiation between the probabilities of possible bundle transmissions. Moreover, it can later be supplemented by additional dynamic context information such as the average contact duration and contact probability which are determined from recorded contacts.

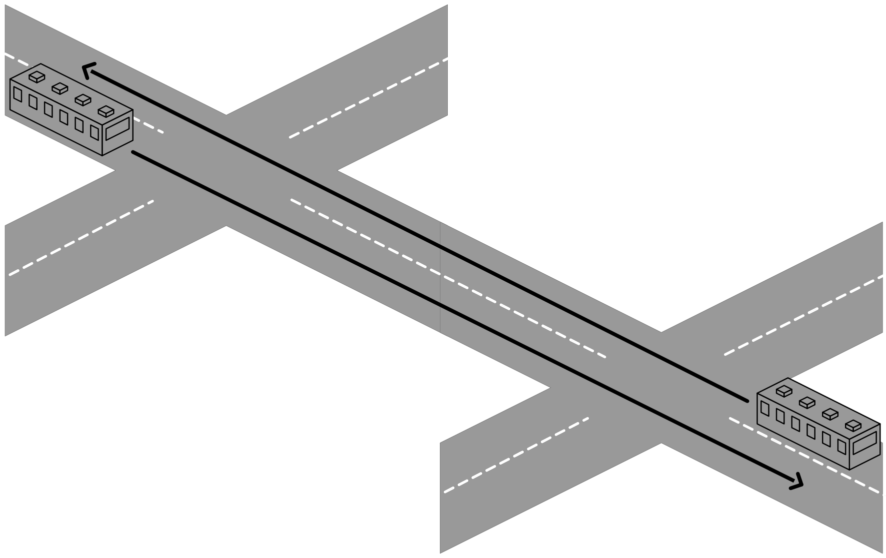

Figure 5.3: Bundle transmission at an overlapping.

5.3.5 Extensions

We introduce further extensions to make the routing procedure more efficient. At the beginning of each contact, announcement messages are exchanged to identify the set of bundles already buffered on the other node. After bundle copies are distributed, it may happen in the case of unpredicted disturbances that two nodes making contact hold the same bundle. Instead of transferring the bundle again, a meta-bundle containing an update message is transferred. This updates the number of bundle copies (n) on the other node and makes the transmission of the whole bundle unnecessary. Moreover, a node is informed of successfully delivered bundles. For this, a flooding mechanism is used with a hop-limit. The announcement message is also used to identify a vehicle and inform other nodes about the current state of a vehicle, e.g. its direction and delay.

A priority can be assigned to each bundle and determines the number of bundle copies and the transfer order. The transmission of bundles is in the order of priorities and within the priorities it is ordered by the bundle age. The bundle age is in ascending order to allow a delivery as fast as possible and to release resources as soon as possible.

Further optimizations are for the delivery to a gateway. This is worthwhile because the probability of a successful delivery is very high when the gateway is on the path of a vehicle. Only heavy disturbances can prevent the delivery because the number of hops is minimal. Therefore, one bundle copy is kept permanently on the node until it makes direct contact with the gateway. Often this is not the fastest way to transfer a bundle, but it is more reliable and saves resources. All other bundle copies are sent as supplied before and may reach the destination faster but with higher uncertainty.

5.3.6 Implementation

We have implemented our routing protocol for ‘The ONE’[1] [74], a DTN simulator featuring several previously proposed routing protocols that we use for performance comparison. For RUTS, it is necessary to make some context information accessible on every DTN node, such as the network map and timetables. The timetables or other dynamic context information are stored in a database. For the network map and the routing-graph, the free Java Graph Library ‘JGraphT’[2] is used and allows for simple management of the graphs. Furthermore, it included a simple search for the k-shortest ways and a check for cycles in the graph.

The routing paths are evaluated to calculate the score. For this purpose, the possible bundle transmissions are analyzed and rated. It is necessary to determine the type and quality of the encounter in consideration of the directions of each vehicle. This is provided through an evaluation of context information. For the management of the number of copies and the priorities of a bundle, the simulator allows to define additional fields in a bundle so that they can be used for this purpose. Further functionality has been added to update the number of bundle copies before it is transmitted and to restore it if a transfer fails. Finally, in addition to the data bundles, the announcement and update bundles are needed. Again, the available additional fields in the bundles have been used to transmit the necessary data.

[1] http://www.netlab.tkk.fi/tutkimus/dtn/theone/
[2] http://jgrapht.sourceforge.net/

5.4 Evaluation

The purpose of the simulation-based evaluation is to analyze the performance and characteristics of the developed routing scheme under realistic conditions. Moreover, the evaluation compares our routing algorithm to previous approaches. As a first step the Braunschweig model is used for various simulation scenarios. Like already described in detail in section 3.2.1.1, this model is synthetic. Nevertheless, it is quite realistic, because node mobility is based on real cartographic data and the SUMO[58] micro-mobility simulator, which also takes traffic lights, multiple lanes and the influence of other vehicles into account. Nevertheless, it is based on idealized system behavior. However, the great advantage of a synthetic model is that it is completely controllable. This means that it is possible to induce consistent disruptions (e.g. a large number of late vehicles due to a traffic jam) into the whole model. On contrary, in a real world mobility trace, only single vehicles can be manipulated, but it is not possible to create a systematic, consistent scenario, because there are no rules that define the interactions, dependencies and side-effects of this manipulation. Therefore, this synthetic model plays an important role in the development and evaluation of routing mechanisms.

However, only an evaluation based on real-world mobility allows for a valid performance analysis. It also shows that the algorithm design does not rely on the idealized behavior of the synthetic model. For this reason the Chicago mobility trace described in section 3.2.1.3 is used in the second step of the evaluation. This trace contains various unexpected properties (as described before) and is much larger than the Braunschweig model. Moreover, the public transport routes follow a grid like layout instead of the "star-with-ring" topology of Braunschweig's public transport network.

For the evaluation, both the Braunschweig model and the Chicago trace are considered in the following scenarios:

1. With the Braunschweig model it is shown that the developed routing scheme works as intended. The simulation results are compared to various existing routing approaches.

2. Then the effect of severe disruptions of the public transport network such as delays and breakdowns of increasing amounts of vehicles are evaluated. Besides these disruptions the scenario remains the same. This allows for a comparison of performance in the non-disrupted case for RUTS as well as for the previously proposed routings.

3. Finally, RUTS is evaluated in several scenarios based on the Chicago mobility trace. Again, it is compared to existing solutions. However, because of the large size of the Chicago scenario, it is not possible to simulate all routings that are implemented for The ONE, since not all implementations (for The ONE) scale well enough. However, optimal solutions are calculated with our own PerfectRouter (using Dijkstra's algorithm and perfect knowledge) in order to determine the upper bound, i.e. the theoretically best possible performance. Moreover, several different scheduling approaches for RUTS are evaluated, in order to find out if there is a

significant potential in the prediction of contact durations that were analyzed in section 4.1.

5.4.1 Performance Metrics

The recognized metrics for DTN routing [97] are not completely congruent with the usual end-to-end [98] metrics, because of the different network paradigms. The most recognized primary metrics for DTN environments are delivery ratio and the latency of successfully delivered messages. These two are also most relevant to user 'experience'. On the other hand, there are secondary performance indicators for scalability (i.e. the suitability for scenarios with increasing amounts of nodes and network traffic), that might not have an impact on user's experience under normal conditions, but only under high load or in a very large network. However, these metrics (e.g. buffer usage or the number of replicated messages) are important to compare the suitability of several routing algorithms in specific scenarios.

Another class of performance metrics is used to evaluate an algorithm's internal behavior, e.g. the average number of hops that are required for delivery. These tertiary metrics are not used in the following evaluation for two main reasons. First, the internal details of the routing algorithms are not useful for a fair comparison, because different approaches implement different behavior. For example the number of hops will be different for a "direct delivery" and a "hot potato" or "first contact" approach. The second reason is that these indicators are redundant, because the primary metrics already describe performance across different algorithms. Therefore, the internal details are not considered in the evaluation of RUTS. Instead, we focus on the following primary and secondary metrics:

- **Delivery ratio**: This is the ratio between the number of created and the number of delivered messages. In simulations this value typically changes over time, especially during the "warm-up" phase in which many messages are created but not yet delivered. Therefore, the delivery ratio is a snapshot at a certain point of time, usually at the end of the simulation. As a single value it is well suited for performance comparisons, but it gains additional informativeness if it is considered in relation with bundles that are still in transit. For this reason two additional indicators are used in the evaluation:
  - **Remaining bundles**: These are bundles that are still in transit. At the point of time at which the delivery ratio was calculated these bundles have not yet been delivered, but still have a chance to be delivered later.
  - **Lost bundles**: These bundles will definitely not be delivered any more. The main reason for loosing bundles is that they (and all remaining replicates, if applicable) have been discarded because their lifetime expired or because of full message buffers on the nodes.
- **Delivery latency**: This is the time that passes between the generation and the successful delivery of a bundle. Its value is influenced by various parameters (e.g. message buffer capacity of the nodes, mobility characteristics, radio range, ...).

Strictly speaking this is a per bundle metric, because the delivery latency fluctuates heavily due to the intrinsic properties of DTNs. Therefore, it is common to use the term “delivery latency” for the *average* latency of all successfully delivered bundles. However, the minimum and maximum latency are often also of interest for the characterization and comparison of different algorithms.

- **Buffer usage**: The message buffer on DTN nodes is limited, and because of the store-carry-forward paradigm it limits the message capacity that a node can carry. If the buffer is full, new messages are either not accepted or other messages have to be discarded to make room. Buffer usage is a secondary metric because it does not necessarily effect user experience (i.e. delivery ratio or latency) as long as no (or only a few) buffers are full. However, if the number of nodes in a DTN increases, buffer capacity per node usually remains constant and the DTN’s total buffer capacity only grows linearly. But with replicating (or flooding) routing the number of messages grows exponentially. Therefore, buffer usage is a good metric for scalability. In general a high buffer utilization indicates that the network is near congestion. (However, full buffers are only a sufficient but not a necessary condition for congestion. Low buffer utilization at congestion indicates that another bottleneck in the network is causing congestion.)

From an user’s or application developer’s point of view mainly the primary metrics delivery ratio and latency are of interest. In order to evaluate and compare the scalability, buffer usage has to be considered. In general all metrics depend on the characteristics of a specific scenario, and a fair comparison is only valid within an identical set of diverse scenarios.

5.4.2 Simulation with the Braunschweig Model

RUTS was originally developed as routing facility for the EMMA system in Braunschweig described in the motivation (see section 2.1.2.2). Therefore, the Braunschweig model plays an important role in the evaluation. The model’s characteristics and the toolchain it uses were already discussed in section 3.2.1.1 in detail. It covers the city center and comprises 28 simultaneously moving vehicles on 13 lines with 54 stops. The public transport network of Braunschweig is laid out as a star topology with a ring.

5.4.2.1 Simulation Setup

The scenario has been simulated for 5 hours. In the first hour the vehicles were successively integrated into the simulation in a realistic way. During this startup period the vehicles are not instantly available but start serving the lines from the terminals. It takes some time until the network is fully populated and for this reason the startup period is ignored in the evaluation. The same applies for the shutdown period which is the last hour of the simulation. This is necessary because RUTS derives DTN routes by “looking into the future” and calculating future vehicle routes based on the public transport timetables.

The vehicular nodes have a local copy of the timetables and the network map. For a connection with a backend system two stationary gateways are available. Furthermore, there is a stationary relay. In contrast to a gateway it has no WAN-connection. Through the usage of relays and gateways, the delivery probability and reactions to external intermittent failures such as breakdowns and delays are improved.

Additional reports for "The ONE" have been implemented to record RUTS specific simulation events. Moreover, adjustments were made to provide a continuous connection between the gateways and to avoid communication between vehicles that are not yet in service. These adjustments were necessary because The ONE version 1.0.3 does not support nodes that are "entering" or "leaving" the simulation area. So in order to emulate inactive vehicles these were placed in a "waiting area" with negative coordinates. All events that involved negative coordinates were later ignored in the reports.

For the data exchange between nodes, the simulator can generate bundles in a constant interval at a random node. The destination node for each bundle is also random. All information on the bundle creation was recorded and reused so that all routing methods are simulated under the same conditions. The simulation scenario is inspired by the EMMA use-case. Therefore, the bundle interval is set to 5 seconds and the bundle size to 100 kB. This causes heavy load, which is interesting because of the high resource requirements. Some of the routing protocols reach their limit already at this bundle interval while others still achieve good results. RUTS was configured to conservative parameters: in addition to the original bundle, two replicates are used. In the routing-graph, five possible paths are determined. Thus, $n = 3$ and $k = 5$.

5.4.2.2 Simulation Results

In the evaluation the above performance metrics are used. Like already discussed, some performance metrics may have an impact on others. However, some routing protocols obtain very good results in certain areas but drop back in other fields. Therefore it is important to consider the set of metrics as a whole. First the results of scenario 1, in which all vehicles are on time, are presented. Then, in scenario 2, we vary the number of vehicles that are delayed. This is for the purpose of determining how robust RUTS is against road traffic disturbances and variations of arrival times, which are not uncommon as the mobility analysis of the Chicago trace has shown.

Scenario 1: All vehicles on Time In figure 5.4 the successfully delivered bundles are illustrated as well as the lost bundles and the bundles remaining in transit at the end of the simulation. These sum up to 100% and are equal to the number of generated bundles. Absolute values are shown in table 5.1. In the comparison of all routing protocols, RUTS can deliver the most bundles and also causes the lowest bundle loss. Also, the number of remaining bundles is low. PRoPHET and Epidemic also have a low number of remaining bundles, which is due to their high loss rate.

Figure 5.5 shows that RUTS occupies lesser storage capacity than nearly all other routing protocols. Only FirstContact takes slightly less memory in our simulation because it uses no additional bundle copies. However, this causes a low delivery ratio, as shown in figure 5.4. All other protocols take significantly more memory resources.

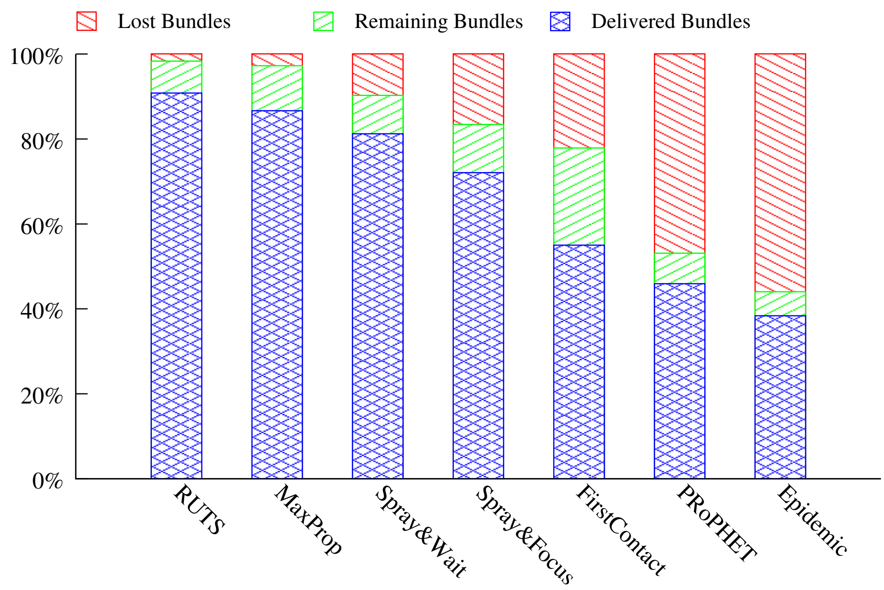

Figure 5.4: Lost, remaining and delivered bundles.

	Lost	Remaining	Delivered
RUTS	37 (2%)	162 (7%)	1961 (91%)
MaxProp	60 (3%)	229 (10%)	1871 (87%)
Spray&Wait	211 (10%)	195 (9%)	1754 (81%)
Spray&Focus	359 (17%)	245 (11%)	1556 (72%)
FirstContact	479 (22%)	494 (23%)	1187 (55%)
PRoPHET	1014 (47%)	155 (7%)	991 (46%)
Epidemic	1210 (56%)	122 (6%)	828 (38%)

Table 5.1: Lost, remaining and delivered bundles.

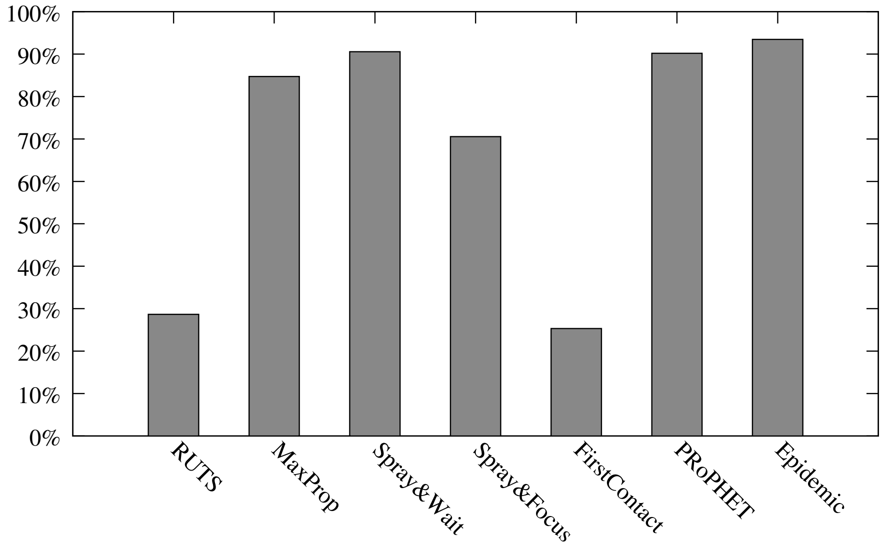

Figure 5.5: Average message buffer usage.

Figure 5.6 and table 5.2 compare the transmission times of successfully delivered bundles. For each simulation run a set of randomly chosen sources with random destinations is calculated. The same set is used for all routing algorithms. Therefore, results are reproducible and comparable. Due to the random selection of the source/destination pairs it is possible that bundles are generated on nodes which are currently in direct contact with the destination. These bundles can be delivered immediately, so that the minimum transmission time is very low. Average and maximum transmission times can only be calculated for bundles which are delivered before the simulation ends. Therefore transmission time can only be interpreted in relation to the number of delivered bundles, because a higher number of delivered bundles increases the probability that bundles with high transmission times are included. In this context RUTS achieves a low average delivery latency, although it has the best delivery ratio.

The results of the simulation for scenario 1 can be summarized as follows:

- RUTS is with the top routers in regards to low average delivery latency (749 s). It's worst case latency is 5825 s, which is in the middle field. However, RUTS reaches the best delivery ratio of 91% and one of the lowest buffer usages.
- MaxProp's delivery latency of 1029 s is quite high, but it delivers the second best number of bundles (87%).
- SprayAndWait is able to perform with a very low latency (706 s) and has the third best delivery ratio of 81%.
- SprayAndFocus does not beat SprayAndWait in this scenario. With a latency of 767 s and a delivery ratio of 72%, its only advantage over SprayAndWait is its lower buffer utilization.

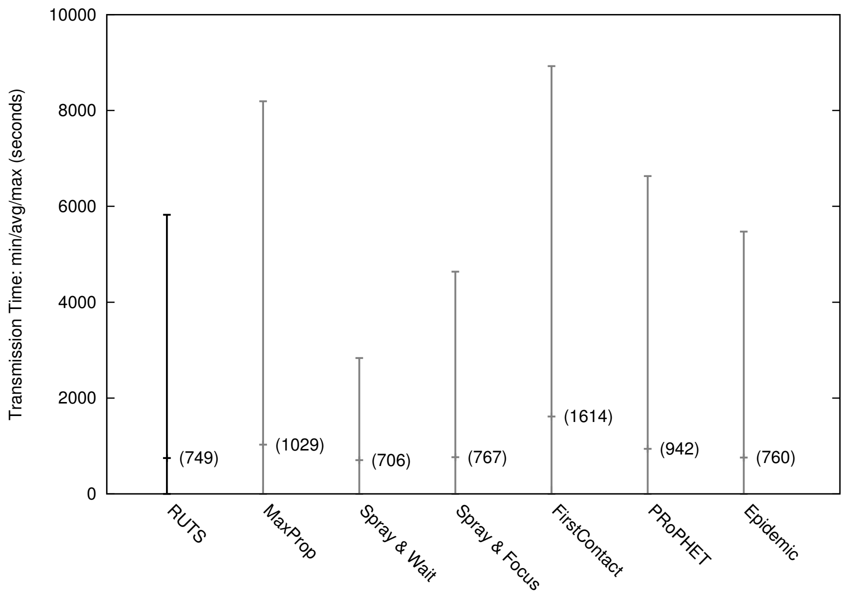

Figure 5.6: Transmission time (Braunschweig trace).

- FirstContact has the worst latency (1614 s) and a low delivery rate of 55%. However, as a matter of principle of its simplistic approach, the buffer usage remains low.
- ProPHET delivers 46% of all bundles with a latency of 942 s, which is in the middle field. In the evaluated scenario it is not able to take full advantage of its probabilistic experience. This is due to the vehicles changing line assignment. However, our mobility analysis of real world traces has shown that this is a property of real-world public transport networks.
- Epidemic achieves a very good latency that is slightly better than RUTS (760 s vs. 794 s), but only delivers 38% bundles. This is due to its excessive replication strategy that does not fit well with the limited amount of resources in the evaluated scenario.

Scenario 2: Vehicle Delays and Vehicle Breakdowns Vehicles may be delayed. To investigate the impact of disturbances in the public transport system several variations to the scenario were generated. 20% of the vehicles are chosen at random and are assigned a random delay of 1-5 minutes. Thereafter the result is calculated as the average of several simulations. The same procedure is used for potential breakdowns of some vehicles. For this purpose, 10, 20, 30 and 50% of the vehicles are chosen at random and removed from the simulation. As a result, no bundles to the destination vehicles can be delivered and hence these remain longer in the network. On the other hand, the removed vehicles can no longer emit any bundles.

Transmission Time (s)	min	max	avg
RUTS	0	5825	794
MaxProp	0	8194	1029
Spray&Wait	0	2834	706
Spray&Focus	0	4673	767
FirstContact	0	8926	1614
PRoPHET	0	6632	942
Epidemic	0	5472	760

Table 5.2: Transmission time (Braunschweig trace).

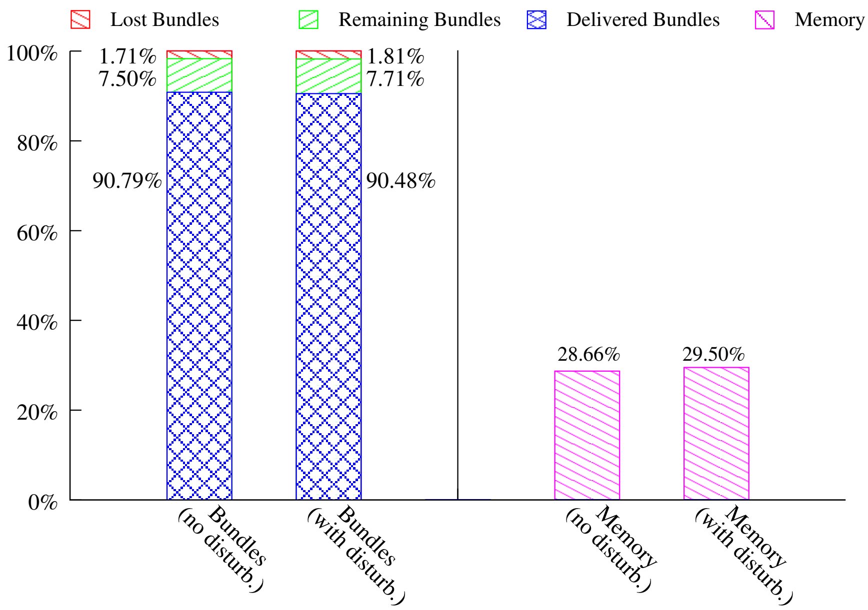

Figure 5.7: Impact of disturbances.

Figure 5.7 shows the differences between the previous results and the average results with random delays. On the left side the results for the delivered, lost and remaining bundles are compared and on the other one the average memory usage is contrasted. As you can see, the differences are very marginal (<1%). This is because RUTS creates a low but sufficient number of bundle duplicates which are routed via another path, so that it is robust against disturbances.

The same also applies for the simulated breakdowns in figure 5.8. The lines converge with an increasing number of removed vehicles. Nevertheless, RUTS successfully delivers the most bundles under all conditions.

The performance of RUTS is very promising when taking into account all performance criteria. In all investigated areas RUTS is at least similar or better than the best performing routing protocol. In most cases it achieves the best results.

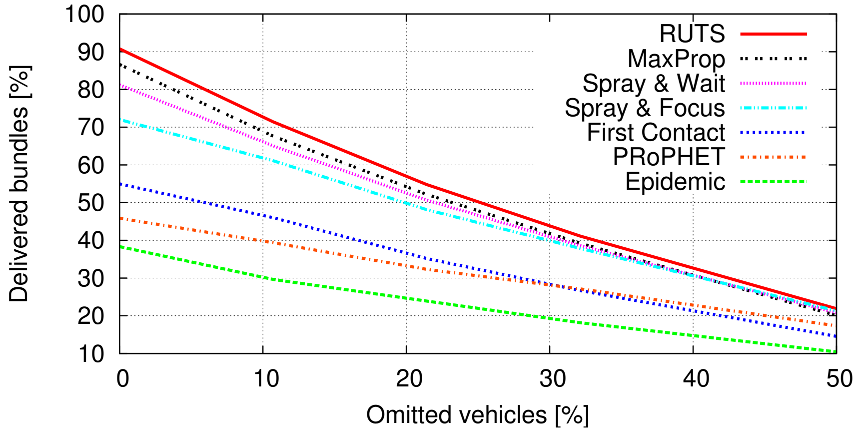

Figure 5.8: Impact of breakdowns.

5.4.2.3 Conclusion

In this section we presented the evaluation of RUTS in the Braunschweig scenario, which shows that low resource utilization, low latency and high delivery rates are achieved by RUTS, and thus are not conflicting objectives for DTN routing. One could argue that RUTS profits only from the known timetables, but the performance decreases only marginally even if 20% of the vehicles are behind schedule or 10% fail completely. Another reservation against RUTS could be its high level of specialization. However, DTNs are deployed in special scenarios. Generic approaches like epidemic routing work in almost any scenario, but the performance is often poor. In our opinion the increased performance of RUTS outweighs the advantage of a generic algorithm.

5.4.3 Experimental Evaluation with the Chicago Trace

In the previous section it was shown that RUTS operates as intended with a very realistic but nevertheless synthetic mobility model. Now, we investigate the performance of RUTS with our large-scale real world mobility trace of Chicago, that was introduced and analyzed in section 3.2.1.3. As discussed before, the public transport of Chicago is one of the largest bus-based systems. In fact, it is so large that simulations with the trace require significant computation and memory resources. In general a single simulation with The ONE 1.3 and realistic DTN traffic patterns needs several months on a current mid-end server (16 cores, 32GB RAM). This is partly due to the high number of connections that the simulator first has to detect by calculating the distance between every pair of nodes. Then the simulator has to check if there are messages to be transferred for each existing connection, which requires stepping through all messages. The sheer number of these steps is already requiring significant time to compute.

But additionally, and even more complex, the routing modules to be evaluated cause their own - much higher - load. This load is not necessarily only caused by routing algorithms that are inherently greedy, but may also be caused by the specific implementation for The ONE. As common for Java code, these implementations often use data structures that are easy to debug but quite resource hungry. Unfortunately, this means that some algorithms that are implemented for The ONE do not scale well enough for simulations with the whole Chicago trace. The typical simulator behavior in this case is to slow down exponentially at a certain, reproducible point of simulation time. This results in simulations progressing so slowly that it is evident that they will not finish in reasonable time (i.e. years or decades). The EpidemicRouter is an example for this behavior[3]. An other typical behavior is that the simulator uses up all available RAM[4] and than crashes (which is not the simulators fault but the greedy implementation of the routing algorithms, which can be shown by using the same setup with PassiveRouter). Assigning swap space is not an option in this case because "virtual memory" is significantly slower. A typical representative of this behavior is the ProphetRouter.

Therefore, in order to enable simulations with the Chicago trace certain compromises have to be made. First, in this part of the evaluation RUTS will only be compared to other routing algorithms that compute in a reasonable time (up to two weeks) and with a reasonable amount of memory (8 GB). This allows for several simulations running in parallel on our simulation server. Moreover, the load is reduced by only using one source node per simulation. To keep the results representative, 50 simulations with different source nodes are computed per scenario and the results are averaged. With this additional limitation the routers that fulfill the above criteria are Spray-and-Wait, First Contact, Epidemic and RUTS. However, it should be kept in mind that the functionality of RUTS was already shown in the previous section, along with a comparison to several more other algorithms. This part of the evaluation primarily focuses on showing that RUTS also performs well with a real mobility trace and with experimentally determined channel characteristics.

5.4.3.1 Methodology

The experimental evaluation with the Chicago trace has a different focus than the simulation with the Braunschweig model. With the model, it was shown that RUTS operates as it was designed within a consistent model. As always, such a model reduces complexity by replacing reality with a set of rules. Now, RUTS will be evaluated with a real mobility trace, which is not a simplified model. The purpose of this part of the evaluation is to show that the design of RUTS does not rely on these simplifications. However, it is explicitly *not* intended to compare quantitative results of the Braunschweig model with the Chicago trace, as both public transport networks are completely different. Moreover, it is beyond the scope of this thesis to quantify the degree of realism that can be reached with a synthetic model. For these reasons both parts of the evaluation are deliberately chosen to complement each other, instead of comparing apples to oranges. In summary the evaluation with the Chicago trace has the following different properties:

- DTN traffic is simulated according to the more bursty use-case "onboard passenger

[3]After 4 weeks of computing on our server it had progressed to around one hour of simulation time. Then it gradually slowed down and took several hundred seconds for one single step (which corresponds to less than 1 ms simulation time). After 12 weeks no progress was observable at all.

[4]In our case we assigned up to 30GB to the Java VM, which was used-up after a few minutes.

information systems" instead of the constant bundle rate like in use-case "air quality monitoring". This also means that bundles are transmitted between changing source/sink pairs instead of being sent from multiple sources to a common sink. Moreover, in this use-case bundles have a variable payload length. Therefore, the number of delivered messages is no longer a suitable metric. Instead, the sum of all delivered payload data will be used.

- The scenario is deliberately chosen to be very demanding. Bundle sizes follow the demands of the use-case and are 10-100 MB. In order to cause heavy bundle load in this scenario we update 100 passenger information systems simultaneously with different content. The updates originate from a mobile node, and no fixed gateways or relays are used.
- There are no artificial delays but only the real delays that originate from the mobility trace and the according real timetables.
- Communication parameters between nodes are set to the values that we obtained in field measurements in a real public transport setup (1587.88 Kbytes/s and 350m range, see section 3.1.2) instead of using the commonly accepted "standard" values.

Again, RUTS will be compared to previously proposed routing algorithms based on the above metrics. However, these are only a subset of the evaluation with the Braunschweig scenario, due to the complex scenario and the limitations that were described in the previous section. Because not all previously proposed routings can be evaluated for above reasons, we give the optimum solution as an upper bound of the (only theoretically possible) best performance. This 'PerfectRouter' is based on Dijkstra's Algorithm and knowledge of all future contacts (which is not possible in a real setup). A detailed description of the PerfectRouter is given in section 4.2.

5.4.3.2 Simulation Setup

According to our previous measurements in a real-world vehicular setup [69], we chose 350m radio range and an average channel capacity of 1587.88 Kbytes/s as realistic parameters. The Chicago mobility trace is integrated as an external movement model into the simulator. As mentioned in the previous section we decided to simulate several subsets in order to gain an acceptable computation time. With this approach a large number of smaller scenarios can be simulated, giving more reliable results than just a few big scenarios. Therefore, smaller scenarios are generated with a subset of 45 nodes and a five hour time window. 50 random pairs of source and sink nodes are selected. According to the use-case "onboard passenger information systems" the message size is uniformly distributed between 10 MB and 100 MB. Moreover, we assume that no more than 100 passenger information systems are updated simultaneously. Therefore, 100 bundles are generated for each of the 50 simulation runs. There are no relay nodes in this setup. At the start of the simulation, bundles are generated at the source nodes, and at the end of the five hour time window The ONE's report files are evaluated. Results for one configuration are given as minimum, maximum and average values of 50 simulation runs with different source/sink pairs.

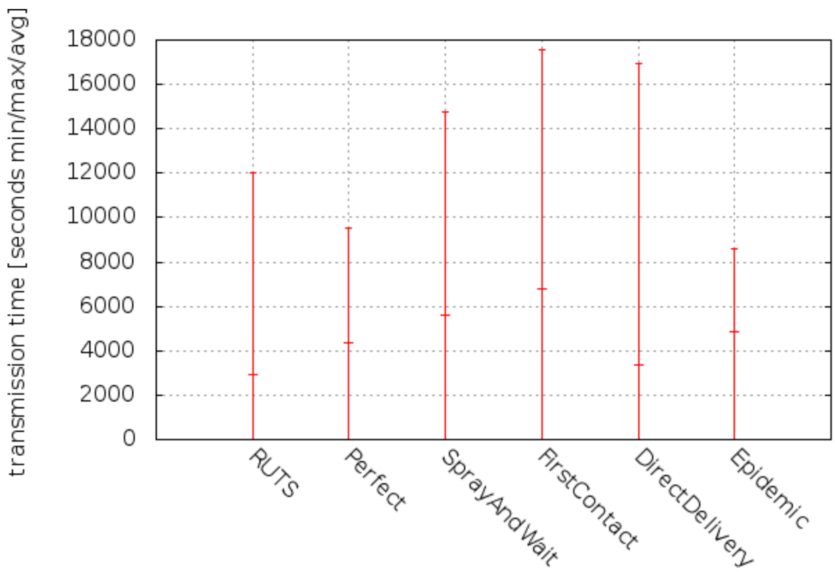

Figure 5.9: Delivery latency (Chicago trace).

5.4.3.3 Simulation Results

Figure 5.9 shows the minimum, average and maximum delivery latency of 50 simulations (minimum of all minima, maximum of all maxima and average of all averages of all simulation runs). A numerical representation is given in table 5.3. First of all, it is notable that the minimum delay is zero for all routers. This is because the random source/sink pairs (the same set of random combinations is used for all routers) contain a pair that is in contact at the beginning of the simulation. A second notable observation is the high maximum delay, especially for FirstContact and DirectDelivery. This means that bundles were still delivered almost at the end of the simulation, which lasts for 18000 seconds. However, these observations also show that the minimum and maximum values are only describing extreme cases. Nevertheless, these values are of interest for best-case and worst-case analysis. A third interesting observation is that the delivery latency of Perfect is not the lowest. This seems unintuitive at a first glace, but is caused because Perfect optimizes for a maximum amount of delivered data, which in turn leads to a much higher channel utilization that causes the higher delivery delays.

But for an use-case-centric comparison the average values are more meaningful. For the interpretation of the results it is important to understand that delivery latency values can only be calculated for delivered bundles. Therefore, the results should be seen in relation with the amount of delivered data which is presented in figure 5.10 and table 5.4. With this background information the performance of some routers that intuitively look very good in figure 5.9 is relativized. Therefore, the following interpretation of the results also takes the total amount of delivered data shown in figure 5.10 into account.

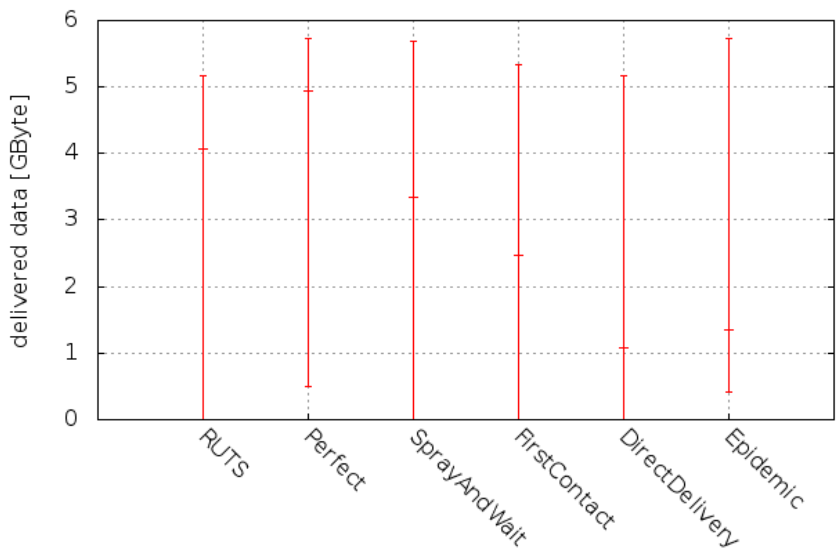

Figure 5.10: Total amount of delivered data (Chicago trace).

The total amount of delivered data is a better metric for this scenario because bundle sizes are dynamic according to the use-case. Simply using the number of bundles would lead to an unfair comparison, because several small bundles can be transmitted in the same time as one large bundle. The maximum total amount of data that is generated is 5.68 GB (with an average of 5.5 GB over all scenarios). The results of the simulation can be interpreted as follows:

- RUTS gets the lowest average latency (2912 s) which is not surprising given the fact that it makes use of vehicle routes and timetables. It also achieves the highest amount of delivered data (4164 MB). Therefore, RUTS has the best performance, although is does not reach the theoretical upper bound of Perfect.
- Perfect is given as an upper bound benchmark. At a first glance it seems to be surprising that it does not reach the lowest latency but only 4370 s. This is due to the omniscient design of Perfect. It makes perfect link scheduling because it knows contacts and contact durations in advance. Therefore it keeps bundles until the best (i.e. network-wide maximum sum of delivered data) route becomes available. As a result it reaches the best possible average amount of delivered data (5065 MB). This value is below the 5500 MB that are generated on average over all of the 50 simulations and shows that the chosen use-case has quite demanding requirements. There are source/sink-combinations that are theoretically unsolvable if a delivery ratio of 100% within 5 hours is required.
- SprayAndWait achieves a medium average latency (5598 s) and delivers a medium

Transmission Time (s)	min	max	med	avg
RUTS	0	11990	2896	2912
Perfect	0	9552	4170	4370
Spray&Wait	0	14747	5018	5598
FirstContact	0	17566	5999	6762
DirectDelivery	0	16926	1130	3367
Epidemic	0	8608	4829	4880

Table 5.3: Transmission time (Chicago trace).

to good amount of data (3592 MB). The maximum values show that SprayAndWait fluctuates heavily, which means that there is an impact of 'good' and 'bad' source/sink pairs on performance.

- FirstContact has a high average latency (6762 s) and a medium amount of data (2532 MB) and also fluctuates strongly. Nevertheless, for its simple approach the performance is quite good compared to the more sophisticated SprayAndWait.
- DirectDelivery only gets a high latency (3367 s) and a low amount of data (1109 MB). This is not surprising, since DirectDelivery simply waits until it is in direct contact with the receiver.
- Epidemic has a medium average latency (4880 s) and the lowest maximum latency (8608 s). It is also the only router (besides omniscient Perfect, which cannot be implemented in the real world) that is able to deliver at least some data in all situations. However, its total amount of delivered data is low (1383 MB). Usually Epidemic achieves low latencies when the network load is low. But in this scenario the load is quite demanding. In combination with Epidemic's inherent replication this leads to a bottleneck in contact capacity, so that latency increases because contacts are not long enough to transfer all bundles. One could argue that the scenario is deliberately disadvantageous for Epidemic because of the high network load. Therefore, we gave Epidemic an additional advantage in this simulation and configured the per node message buffer to 10 GB, which is more than the summed up total size of all bundles. This means that no bundles are discarded or rejected because of full message buffers, and that the only bottleneck results from the limited contact time.

5.4.3.4 Conclusions

In this section RUTS was evaluated with a real mobility trace and an additional realistic use-case. The requirements were quite demanding. First, the computation of the simulations was very complex. Second, the chosen use-case and the evaluated set of

Delivered Data (MB)	min	max	med	avg
RUTS	0	5297	4587	4164
Perfect	512	5862	5181	5065
Spray&Wait	0	5817	3660	3592
FirstContact	0	5470	2595	2532
DirectDelivery	0	5297	588	1109
Epidemic	421	5862	1738	1383

Table 5.4: Total amount of delivered data (Chicago trace).

scenarios cause a network load that exceeds the available capacity for some source/sink combinations. In this setup no router was able to deliver all bundles within 5 hours, which is due to the high network load. However, the evaluation also shows that there is an average capacity of 4164 MB (with RUTS) available. With regard to the use-case this means that an update of 75 (instead of 100) passenger information systems within 5 hours is feasible.

RUTS was compared to several other routing schemes and exceeded their performance significantly. Moreover, it was able to achieve a source-to-sink-throughput that is 82 % of the theoretical upper bound, while the next-best SprayAndWait router only gets 71 % and a slightly higher delivery latency. But the main focus of this part of the evaluation was to show that RUTS performs with a real mobility trace (and real timetables and arrival deviations) and fulfills the requirements of the additional use-case "passenger information systems" even under very demanding conditions. On contrary to the use-case "environmental monitoring" a bursty network load and the resulting overload situation were evaluated. Moreover, it was also shown that RUTS also performs without any relays or gateways.

The comparative evaluation with a real mobility trace also results in some interesting conclusions on the performance of the other routing algorithms. Epidemic's relatively untypical high latency and low amount of delivered data shows that there is not enough capacity for it's extensive replication strategy. DirectDelivery as a matter of principle achieves a very low latency, but at the cost of a very bad delivery ratio.

5.4.4 Evaluation Summary

The evaluation was structured in several steps with increasing realism and complexity. First, a scenario based on the synthetic Braunschweig model was simulated. We compared the performance of RUTS with various other previously proposed routing algorithms. It was shown that the performance of RUTS is significantly better. In the next step the impact of delays and disturbances in the public transport network was investigated. Again, RUTS outperformed the other approaches.

In the next step a real mobility trace and context information of a real public transport network was used. A demanding use-case with an overload-situation was simulated. RUTS achieved a satisfactory performance. It outperformed other routers and reached 82 % of the theoretical optimum solution. Finally, the impact of different scheduling strategies in the same scenario was evaluated. It was shown that the potential performance gain is only marginal.

In conclusion RUTS is able to outperform various existing routers. It is also able to cope with disturbances and overload situations.

Chapter 6

Conclusions and Outlook

Vehicular communication systems are a key technology for user-friendly and efficient operation of public transport. For many years, voice-centric real-time human interaction via analog radios has been used. Now, modern ICT operation support systems require data-centric and often asynchronous machine-to-machine communication to realize inter-networked applications such as mobile passenger information systems, demand-driven dispatching, and remote vehicle monitoring. Although it is generally accepted that legacy communication systems are unable to fulfill the ever increasing capacity requirements for data transfer, many public transport operators are delaying or avoiding migration to newer communication systems because of the high investments and/or operational expenses. License exempt and well established technologies such as IEEE 802.11-based WLAN would be an economic alternative, but has an insufficient radio range. Therefore, a continuous connection of all vehicles is not possible. Installing a separate, city-wide WLAN with complete coverage for public transport, requires a huge number of access points or road side units, and is therefore not an option for cost reasons. However, the concept of disruption tolerant networking with its store-carry-forward approach instead of end-to-end connectivity can solve this problem. We investigated and confirmed the feasibility of such a communication system.

6.1 Contributions

In this thesis a communication system based on DTN and license exempt wireless technologies was developed and evaluated. Thereby, infrastructure components become mostly dispensable, so that costs are reduced. Moreover, communication robustness is increased because of the inherent resilience to disruptions. Our core contributions are in fields of

1. feasibility investigation based on experimental evaluation,
2. trace acquisition and mobility analysis, and
3. proposal and comparative evaluation of a novel routing approach.

6.1.1 Feasibility Investigation

For the feasibility investigation realistic use-cases and their requirements were introduced. In several measurement campaigns we evaluated IEEE 802.11 a and b (as representatives for current license exempt technologies) for disrupted vehicular communication in urban areas that are typical for public transport routes.

The particular contributions and results in this field are:

- the experimental determination of signal strength, maximum range, transmission throughput, and the influence of mobility and vehicle speed;
- the investigation of the impact of non-line-of-sight and multipath propagation due to urban structures;
- a use-case driven proof of suitability of WLAN technologies for vehicular DTNs (sufficient performance); and
- tangible, numeric measurement results to be used for system design and as parameters for simulation.

It was shown that these technologies are a solid foundation for a disruption tolerant communication system in public transport. To the best of our knowledge such an extensive feasibility investigation was not conducted in previous work.

6.1.2 Trace Acquisition and Mobility Analysis

Several mobility traces were analyzed in order to discover and describe special properties of node movement in public transport networks. Moreover, we acquired a new large-scale trace of the public transport system of Chicago. Our new trace contains additional meta-data on public transport operations. With this meta-data we have shown that there are unexpected properties, e.g. the frequent change of vehicle-to-line assignment.

Furthermore, we investigated the influence of radio range on the duration of different classes of contacts. The resulting distributions are valuable tools for estimating capacity for a given range. We have shown that the different situations in which contacts occur lead to significantly different average durations, and that relatively simple criteria are sufficient to asses these situations.

Our particular contributions are:

- A new, publicly available [1] [68], real-world large scale trace for development and evaluation of public transport DTNs;
- the analysis and description of mobility characteristics as well as public transport specific properties;
- development of criteria and a classification of situation specific inter-vehicle contacts;
- a simulation-based investigation of different scheduling approaches; and
- simulation results that allow a tangible estimation of contact duration distributions at given radio ranges.

[1] http://www.ibr.cs.tu-bs.de/users/mdoering/bustraces/

Besides our Chicago trace, there are - to the best of our knowledge - no other traces publicly available that provide a similar amount of position- and meta-data. Moreover, an extensive analysis of situation specific contact characteristics was not published before ours in [69].

6.1.3 Novel Routing Approach

We proposed a routing scheme that uses special properties of public transport networks and is able to outperform previous approaches. Our comparative performance evaluation has shown that it is resilient against severe disturbances. Moreover, with the evaluation we made the final step to show that the overall communication system is feasible and fulfills the requirements of practical use-cases.

The proposed routing approach is optimized for public transport and exploits context information such as vehicle routes and timetables. Route candidates are weighted with expected delay and contact probability by the source. In case of failed contacts (e.g. due to road disturbances) the route is recalculated on the current node. This results in a high resilience against mobility deviations, as shown in the evaluation, which also shows that our approach outperforms previously proposed routing schemes.

In the field of routing and evaluation we made the following particular contributions:

- a proposal of a novel routing scheme based on the results of our real-world mobility analysis;
- a comparative evaluation with various existing routing algorithms, with simulation scenarios based on realistic use-cases and real-world mobility traces;
- the investigation of the influence of disturbances and breakdowns on our routing; and
- the proof-of-concept implementation of our routing that outperforms existing approaches.

Our comparative evaluation is the final step in showing that the envisioned DTN-based communication system is feasible. To the best of our knowledge such an extensive and holistic feasibility study of WLAN-based DTN in public transport was not conducted before.

6.2 Outlook

Although this thesis is focused on investigating feasibility in a realistic way and with as much real data as possible, the final real-world implementation of the system is still work in progress and goes far beyond the scope of this work. Nevertheless, there are already some prototypical installations[2], which were implemented and evaluated in cooperation with a supplier of public transport equipment.

[2] see the OpTraCom project: http://www.optracom.de

Besides the general task of integrating public transport specific hardware and applications, as well as deploying a larger real-world installation, there are still several areas for interesting extensions. Some general examples are:

- **Integration of smartphones** carried by public transport users, e.g. with passenger information applications. This could be implemented either by a regular access point in the vehicle, or by a smartphone app that operates as a full featured DTN node. In this case an incentive system could be used to motivate passengers to participate (e.g., miles for bandwidth).
- A **hybrid DTN/3G approach** in which DTN nodes are extended with 3G connections that are used for expedited (and expensive) data transmissions, while non-time critical data is transmitted by DTN. Moreover, this could also include 3G offloading by DTN, thereby creating a win-win situation for the operators of both networks.
- **Dual channel ISM radios** with both long-range-low-bandwidth (e.g. long range adaptations of IEEE 802.15.4) and short-range-high-bandwidth capabilities. This configuration would offer the advantages of both technologies. Thereby, the latency of typical signaling messages (small but urgent) would be significantly decreased.

In the field of routing there are also some interesting extensions to investigate:

- **Adaptive replication** that increases the number of replicates if severe disturbances are detected. If several planned contacts have failed, it could be advantageous to use a more aggressive replication strategy. On the other hand, this might provoke congestion. However, the utilization of message buffers could be used in a utility function, that replicates more conservatively at high load and more aggressively at failing contacts. As this would cause some dynamic control loops, a careful and thorough simulation-based investigation is required.
- More **accurate path weighting with cartographic data** (e.g., OpenStreetMaps) instead of vehicle route segments. The current implementation uses busstops to identify route segments. The more segments overlap the higher the probability of a successful contact, because long overlapping segments decrease the influence of arrival deviations. Using the actual road length would give more accurate results. However, this does not necessarily result in better routing performance, so a closer examination and evaluation is necessary.
- **Self learning or success feedback** capabilities could be used to identify locations in which successful contacts are below or above average. Again, this would result in a dynamic behavior and requires a good understanding of control theory as well as extensive simulations.

However, it should be kept in mind that these extensions also increase complexity. From our experience during the development of the routing algorithm it seems that simpler solutions are more likely to deliver stable results. Moreover, higher complexity also makes cause-and-effect relationships much harder to understand. For this reason an evaluation with real mobility data is essential for the evaluation of the proposed extensions.

Bibliography

[1] BBR Verkehrstechnik, "private communications."

[2] D. Graham-Rowe, "Electronic paper rewrites the rulebook for displays," *Nature Photonics 1, 248 - 251 (2007) doi:10.1038/nphoton.2007.53*, vol. 1, pp. 248 – 251, 5 2007.

[3] K. Ito, S. F. D. Leon, and M. Lippmann, "Associations between ozone and daily mortality: Analysis and meta-analysis," *Epidemiology*, vol. 16, pp. 446–457, Jul 2005. Published by: Lippincott Williams & Wilkins.

[4] M. Lippmann, K. Ito, A. Nadas, and R. Burnett, "Association of particulate matter components with daily mortality and morbidity in urban populations." Research report (Health Effects Institute), Aug 2000. (95):5-72.

[5] M. L. Bell, D. L. Davis, and T. Fletcher, "A retrospective assessment of mortality from the london smog episode of 1952: The role of in fluenza and pollution," in *Urban Ecology* (J. M. Marzluff, E. Shulenberger, W. Endlicher, M. Alberti, G. Br adley, C. Ryan, U. Simon, and C. ZumBrunnen, eds.), pp. 263–268, Springer US, 2008.

[6] The European Parliament, The Council of the European Union, "Directive 2000/69/EC of the European Parliament and of the Council of 16 November 2000 relating to limit values for benzene and carbon monoxide in ambient air." Official Journal L 313 , 13/12/2000 P. 0012 - 0021, November 2000.

[7] The Council of the European Union, "Council Directive 96/62/EC of 27 September 1996 on ambient air quality assessment and management." Official Journal L 296 , 21/11/1996 P. 0055 - 0063, September 1996.

[8] The Council of the European Union, "Council Directive 1999/30/EC of 22 April 1999 relating to limit values for sulphur dioxide, nitrogen dioxide and oxides of nitrogen, particulate matter and lead in ambient air." Official Journal L 313 , 13/12/2000 P. 0012 - 0021, April 1999.

[9] The European Parliament, The Council of the European Union, "Directive 2002/3/EC of the European Parliament and of the Council of 12 February 2002 relating to ozone in ambient air." Official Journal L 067 , 09/03/2002 P. 0014 - 0030, Feburary 2002.

[10] S. Poduri, A. Nimkar, and G. S. Sukhatme, "Visibility monitoring using mobile phones." http://robotics.usc.edu/ mobilesensing/visibility/MobileAirQualitySensing.pdf, Dec 2009.

[11] E. J. McCartney, *Optics of the Atmosphere: Scattering by Molecules and Particles, Wiley series in pure and applied optics.* Wiley, New York, 1976.

[12] P. Dutta, P. M. Aoki, N. Kumar, A. M. Mainwaring, C. Myers, W. Willett, and A. Woodruff, "Common sense: participatory urban sensing using a network of handheld air quality monitors," in *SenSys* (D. E. Culler, J. Liu, and M. Welsh, eds.), pp. 349–350, ACM, 2009.

[13] P. M. Aoki, R. J. Honicky, A. Mainwaring, C. Myers, E. Paulos, S. Subramanian, and A. Woodruff, "Common sense: Mobile environmental sensing platforms to support community action and citizen science," in *UbiComp* (H. Y. Youn and W.-D. Cho, eds.), vol. 344 of *ACM International Conference Proceeding Series*, pp. 59–60, ACM, Sep 2008.

[14] S. Lahde, M. Doering, W.-B. Pöttner, and G. Lammert, "Mobile and distributed measurement of air pollution in metropolitan areas using car2x techniques," in *Proceedings of the 3rd Symposium on Informationssysteme für mobile Anwendungen (IMA)*, GZVB, October 2006.

[15] Stadt Braunschweig – Der Oberstadtdirektor, *Dynamisches Schadstoffkataster – INFO-REGIO Teilprojekt 4 – Abschlussbericht der Stadt Braunschweig.* No. Heft 17 in Schriftenreihe Kommunaler Umweltschutz, Stadt Braunschweig, January 2001. [german].

[16] R. Rinke, A. Wieser, B. Vogel, U. Corsmeier, and C. Kottmeier, "Messungen der räumlichen variabilitäat der luftqualität in einem ballungsraum mittels einer straßenbahn - progress report 2010 for bwplus project," tech. rep., Karlsruhe Institute of Technology, 2010.

[17] S. Lahde, M. Doering, W.-B. Pöttner, G. Lammert, and L. Wolf, "A practical analysis of communication characteristics for mobile and distributed pollution measurements on the road," *Wireless Communications and Mobile Computing*, vol. 7, pp. 1209–1218, Dec. 2007.

[18] A. Vahdat and D. Becker, "Epidemic Routing for Partially-Connected Ad Hoc Networks," tech. rep., Department of Computer Science, Duke University, 2000.

[19] Google Inc., "Google Maps, [URL] http://maps.google.com," August 2006.

[20] V. Cerf et al, "Delay-Tolerant Network Architecture ." IRTF Internet Draft, draft-irtf-dtnrg-arch-05.txt, work in progress, March 2006.

[21] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks," in *Proceedings of the 25th IEEE International Conference on Computer Communications*, INFOCOM '06, pp. 1–11, IEEE, 2006.

[22] W. Zhao, Y. Chen, M. Ammar, M.D. Corner, B.N. Levine, and E. Zegura, "Capacity Enhancement using Throwboxes in DTNs," in *Proc. of IEEE Intl Conf on Mobile Ad hoc and Sensor Systems (MASS)*, October 2006.

[23] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *Proc. of MobiHoc '04*, (New York, NY, USA), pp. 187–198, ACM Press, 2004.

[24] W. Zhao, M. Ammar, and E. Zegura, "Controlling the mobility of multiple data transport ferries in a delay-tolerant network," in *Proc. of INFOCOM 2005*, vol. 2, pp. 1407 – 1418, IEEE, March 2005.

[25] A. Seth, D. Kroeker, M. Zaharia, S. Guo, and S. Keshav, "Low-cost communication for rural internet kiosks using mechanical backhaul," in *Proc. of MobiCom '06*, (New York, NY, USA), pp. 334–345, ACM Press, 2006.

[26] A. Pentland, R. Fletcher, and A. Hasson, "Daknet: rethinking connectivity in developing nations," *IEEE Computer*, vol. 37, no. 1, pp. 78–83, 2004.

[27] C. Holman, K. Harras, and K. Almeroth, "A proactive data bundling system for intermittent mobile connections," in *Proc. of IEEE International Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, IEEE, September 2006.

[28] K. Harras, K. Almeroth, and E. Belding-Royer, "Delay tolerant mobile networks (dtmns): Controlled flooding schemes in sparse mobile networks," in *Proc. of Netwoking 2005*, (Berlin / Heidelberg), pp. 1180–1192, Springer, May 2005.

[29] J. Ott and D. Kutscher, "A disconnection-tolerant transport for drive-thru internet environments," in *Proc. of INFOCOM 2005*, vol. 3, pp. 1849–1862, 2005.

[30] V. Bychkovsky, B. Hull, A. Miu, H. Balakrishnan, and S. Madden, "A measurement study of vehicular internet access using in situ wi-fi networks," in *Proc. of MobiCom '06*, (New York, NY, USA), pp. 50–61, ACM Press, 2006.

[31] R. Gass, J. Scott, and C. Diot, "Measurements of in-motion 802.11 networking," in *Proc. of WMCSA '06*, (Washington, DC, USA), pp. 69–74, IEEE Computer Society, 2006.

[32] M. Doering, S. Rottmann, and L. Wolf, "Design and implementation of a low-power energy management module with emergency reserve for solar powered dtn-nodes," in *Proceedings of the 3rd Extreme Conference of Communication (ExtremeCom 2011), Manaus, Brazil*, 9 2011.

[33] M. Huang, S. Chen, Y. Zhu, B. Xu, and Y. Wang, "Topology control for time-evolving and predictable delay-tolerant networks," in *8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS 2011)*, (Valencia, Spain), October 2011.

[34] Y. Cao and Z. Sun, "Routing in delay/disruption tolerant networks: A taxonomy, survey and challenges," *Communications Surveys Tutorials, IEEE*, vol. PP, no. 99, pp. 1 –24, 2012.

[35] S. Jain, K. Fall, and R. Patra, “Routing in a delay tolerant network,” in *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, (New York, NY, USA), pp. 145–158, ACM, 2004.

[36] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Spray and wait: an efficient routing scheme for intermittently connected mobile networks,” in *WDTN '05: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, (New York, NY, USA), pp. 252–259, ACM, 2005.

[37] T. Spyropoulos, K. Psounis, and C. Raghavendra, “Spray and focus: Efficient mobility-assisted routing for heterogeneous and correlated mobility,” in *Pervasive Computing and Communications Workshops, 2007. PerCom Workshops'07. Fifth Annual IEEE International Conference on*, pp. 79–85, 2007.

[38] A. Lindgren, A. Doria, and O. Schelén, “Probabilistic routing in intermittently connected networks,” *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 7, no. 3, pp. 19–20, 2003.

[39] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, “Maxprop: Routing for vehicle-based disruption-tolerant networks,” in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pp. 1–11, 2006.

[40] C. Liu and J. Wu, “Routing in a cyclic mobispace.,” in *MobiHoc* (X. Jia, N. B. Shroff, and P.-J. Wan, eds.), pp. 351–360, ACM, 2008.

[41] J. Leguay, T. Friedman, and V. Conan, “Dtn routing in a mobility pattern space,” in *WDTN '05: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, (New York, NY, USA), pp. 276–283, ACM, 2005.

[42] J. Leguay, T. Friedman, and V. Conan, “Evaluating mobility pattern space routing for dtns,” in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pp. 1 –10, April 2006.

[43] C. Liu, J. Wu, and I. Cardei, “Message forwarding in cyclic mobispace:the multi-copy case,” in *Proceedings of The 6th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS 2009)*, 2009.

[44] I. Cardei, C. Liu, J. Wu, and Q. Yuan, “Dtn routing with probabilistic trajectory prediction,” in *WASA '08: Proceedings of the Third International Conference on Wireless Algorithms, Systems, and Applications*, (Berlin, Heidelberg), pp. 40–51, Springer-Verlag, 2008.

[45] P.-C. Cheng, K. Lee, M. Gerla, and J. Härri, “Geodtn+nav: Geographic dtn routing with navigator prediction for urban vehicular environments,” *Mobile Networks and Applications*, 2009.

[46] I. Leontiadis and C. Mascolo, “Geopps: Geographical opportunistic routing for vehicular networks,” in *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, pp. 1–6, 2007.

[47] I. Leontiadis, P. Costa, and C. Mascolo, "Extending access point connectivity through opportunistic routing in vehicular networks," in *INFOCOM'10: Proceedings of the 29th conference on Information communications*, (Piscataway, NJ, USA), pp. 486–490, IEEE Press, 2010.

[48] I. Rhee, M. Shin, S. Hong, K. Lee, and S. Chong, "On the levy-walk nature of human mobility," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, (Phoenix, AZ), pp. 924 –932, April 2008.

[49] A. Einstein, "Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen," *Annalen der Physik, Berlin*, vol. 17, pp. 549–560, 1905.

[50] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, MobiCom '98, (New York, NY, USA), pp. 85–97, ACM, 1998.

[51] E. Royer, P. Melliar-Smith, and L. Moser, "An analysis of the optimum node density for ad hoc mobile networks," in *Communications, 2001. ICC 2001. IEEE International Conference on*, vol. 3, pp. 857 –861 vol.3, 2001.

[52] Z. Haas, "A new routing protocol for the reconfigurable wireless networks," in *Universal Personal Communications Record, 1997. Conference Record., 1997 IEEE 6th International Conference on*, vol. 2, pp. 562 –566 vol.2, Oct. 1997.

[53] B. Liang and Z. Haas, "Predictive distance-based mobility management for pcs networks," in *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, pp. 1377 –1384 vol.3, Mar. 1999.

[54] M. Sánchez and P. Manzoni, "Anejos: a java based simulator for ad hoc networks," *Future Gener. Comput. Syst.*, vol. 17, pp. 573–583, March 2001.

[55] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, "A group mobility model for ad hoc wireless networks," in *Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, MSWiM '99, (New York, NY, USA), pp. 53–60, ACM, 1999.

[56] V. A. Davies, "Evaluating mobility models within an ad hoc network," Master's thesis, Colorado School of Mines, 2000.

[57] F. Ekman, "Mobility models for mobile ad hoc network simulations," Master's thesis, Helsinki University of Technology, Department of Communications and Networking, may 2008.

[58] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner, "Sumo (simulation of urban mobility) - an open-source traffic simulation," in *Proceedings of the 4th Middle East Symposium on Simulation and Modelling (MESM20002)*, (Sharjah (United

Arab Emirates).), pp. 183–187., 4th Middle East Symposium on Simulation and Modelling, 9 2002.

[59] S. Krauß, *Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics.* PhD thesis, Mathematisches Institut, Universität zu Köln, 1998.

[60] V. Naumov, R. Baumann, and T. Gross, “An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces,” in *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc ’06, (New York, NY, USA), pp. 108–119, ACM, 2006.

[61] W. jen Hsu, T. Spyropoulos, K. Psounis, and A. Helmy, “Modeling time-variant user mobility in wireless mobile networks,” in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pp. 758 –766, May 2007.

[62] M. Musolesi and C. Mascolo, “A community based mobility model for ad hoc network research,” in *Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality*, REALMAN ’06, (New York, NY, USA), pp. 31–38, ACM, 2006.

[63] H.-Y. Huang, P.-E. Luo, M. Li, D. Li, X. Li, W. Shu, and M.-Y. Wu, “Performance Evaluation of SUVnet With Real-Time Traffic Data,” *Vehicular Technology, IEEE Transactions on*, vol. 56, pp. 3381–3396, Nov. 2007.

[64] X. Zhang, J. Kurose, B. N. Levine, D. Towsley, and H. Zhang, “Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing,” in *MobiCom ’07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, (New York, NY, USA), pp. 195–206, ACM, 2007.

[65] J. G. Jetcheva, Y. C. Hu, S. Palchaudhuri, A. K. Saha, and D. B. Johnson, “Design and evaluation of a metropolitan area multitier wireless ad hoc network architecture,” in *Mobile Computing Systems and Applications, 2003. Proceedings. Fifth IEEE Workshop on*, pp. 32–43, 2003.

[66] S. Ahmed and S. S. Kanhere, “Cluster-based Forwarding in Delay Tolerant Public Transport Networks,” in *LCN ’07: Proceedings of the 32nd IEEE Conference on Local Computer Networks*, (Washington, DC, USA), pp. 625–634, IEEE Computer Society, 2007.

[67] J. G. Jetcheva, Y.-C. Hu, S. PalChaudhuri, A. K. Saha, and D. B. Johnson, “CRAWDAD data set rice ad hoc city (v. 2003-09-11).” Downloaded from http://crawdad.cs.dartmouth.edu/rice/ad_hoc_city, Sept. 2003.

[68] M. Doering, T. Pögel, W.-B. Pöttner, and L. C. Wolf, “A new mobility trace for realistic large-scale simulation of bus-based DTNs,” in *ACM MobiCom 2010 Workshop on Challenged Networks (CHANTS 2010)*, (Chicago, USA), 9 2010.

[69] M. Doering, W.-B. Pöttner, T. Pögel, and L. Wolf, “Impact of radio range on contact characteristics in bus-based delay tolerant networks,” in *Eighth International*

Conference on Wireless On-Demand Network Systems and Services (WONS 2011), (Bardonecchia, Italy), pp. 195–202, 1 2011.

[70] Ubiquiti Networks, "XTREMERange5 TECHNICAL SPECIFICATIONS."

[71] Nicolas Richasse, "Iperf." http://sourceforge.net/projects/iperf/, 15.02.2011.

[72] J. LeBrun and C.-N. Chuah, "CRAWDAD data set ucdavis/unitrans (v. 2006-11-01)." Downloaded from http://crawdad.cs.dartmouth.edu/ucdavis/unitrans, Nov. 2006.

[73] G. Lammert, "Simulation and evaluation of communication performance in bus networks," Diplomarbeit, Technische Universität Braunschweig, 2007.

[74] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation," in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, SIMUTools '09, pp. 55:1–55:10, ICST, 2009.

[75] Braunschweiger Verkehrs-AG, "Liniennetzplan Braunschweig Innenstadtbereich," 2008.

[76] D. J. Dailey, "Smarttrek: A model deployment initiative," technical report Report No. WA-RD 505.1, Washington State Transportation Center (TRAC), University of Washington, May 2001.

[77] Management Information and Transit Technology Section, Transit Division, "Regional smart bus demonstration project evaluation report - september 2002," evaluation report, King County Department of Transportation, September 2002.

[78] R. J. Zygowicz, E. A. Beimborn, Z.-R. Peng, and S. Octania, "State of the art in automatic vehicle location systems," tech. rep., Center For Urban Transportation Studies at University of Wisconsin - Milwaukee, feb 1998.

[79] Chicago Transit Authority, "Bus Tracker API." http://www.transitchicago.com/developers/bustracker.aspx, 15.02.2011.

[80] NIMA - National Imagery and Mapping Agency, "Department of defense world geodetic system 1984," Technical Report TR 8350.2, NIMA - National Imagery and Mapping Agency, Geodesy and Geophysics Department, St. Louis, MO, jan 2000.

[81] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on opportunistic forwarding algorithms," *IEEE Transactions on Mobile Computing*, vol. 6, no. 6, pp. 606–620, 2007.

[82] T. Karagiannis, J.-Y. Le Boudec, and M. Vojnovici and, "Power law and exponential decay of intercontact times between mobile devices," *Mobile Computing, IEEE Transactions on*, vol. 9, pp. 1377 –1390, oct. 2010.

[83] V. Conan, J. Leguay, and T. Friedman, "Characterizing pairwise inter-contact patterns in delay tolerant networks," in *Autonomics '07: Proceedings of the 1st international conference on Autonomic computing and communication systems*,

(ICST, Brussels, Belgium, Belgium), pp. 1–9, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007.

[84] L. D. Hartog, T. Spyropoulos, and F. Legendre, “Using Public Transportation as a DTN Backbone: Mobility Properties and Performance Analysis,” in *Proceedings of IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications*, AOC ’10, June 2010.

[85] J. Ott and D. Kutscher, “Exploiting regular hot-spots for drive-thru internet,” in *Proceedings of KiVS 2005*, Mar. 2005.

[86] D. Ellard and D. Brown, “DTN IP Neighbor Discovery (IPND),” *Internet-Draft*, 2010.

[87] A. Lindgren and K. Phanse, “Evaluation of queueing policies and forwarding strategies for routing in intermittently connected networks,” in *Communication System Software and Middleware, 2006. Comsware 2006. First International Conference on*, pp. 1 –10, 0-0 2006.

[88] B. H. Bloom, “Space/time trade-offs in hash coding with allowable errors,” *Commun. ACM*, vol. 13, pp. 422–426, July 1970.

[89] K. Scott and S. Burleigh, “Bundle Protocol Specification.” RFC 5050 (Experimental), Nov. 2007.

[90] M. Bender, S. Michel, P. Triantafillou, G. Weikum, and C. Zimmer, “Improving collection selection with overlap awareness in p2p search engines,” in *SIGIR* (R. A. Baeza-Yates, N. Ziviani, G. Marchionini, A. Moffat, and J. Tait, eds.), pp. 67–74, ACM, 2005.

[91] D. Hay and P. Giaccone, “Optimal routing and scheduling for deterministic delay tolerant networks,” in *Wireless On-Demand Network Systems and Services, 2009. WONS 2009. Sixth International Conference on*, pp. 27 –34, feb. 2009.

[92] A. Krifa, C. Baraka, and T. Spyropoulos, “Optimal buffer management policies for delay tolerant networks,” in *Sensor, Mesh and Ad Hoc Communications and Networks, 2008. SECON ’08. 5th Annual IEEE Communications Society Conference on*, pp. 260 –268, june 2008.

[93] A. Lindgren, A. Doria, E. Davies, and S. Grasic, “Probabilistic routing protocol for intermittently connected networks (internet-draft draft-irtf-dtnrg-prophet-09, dtn research group).” http://tools.ietf.org/html/draft-irtf-dtnrg-prophet-09, 4 2011.

[94] Y. Xian, C.-T. Huang, and J. Cobb, “Look-ahead routing and message scheduling in delay-tolerant networks,” in *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*, pp. 40 –47, oct. 2010.

[95] S. Merugu, M. H. Ammar, and E. W. Zegura, “Routing in space and time in networks with predictable mobility,” Technical Report GIT-CC-04-07, College of

Computing, Georgia Institute of Technology, 801 Atlantic Drive, Atlanta, GA 30332, mar 2004.

[96] T. Pögel, *Entwickung und Analyse eines DTN-Routingverfahrens für ÖPNV-Netz.* Diploma thesis, Diplomarbeit - TU Braunschweig, 9 2009.

[97] T. Spyropoulos, R. Rais, T. Turletti, K. Obraczka, and A. Vasilakos, "Routing for disruption tolerant networks: taxonomy and design," *Wireless Networks*, pp. 1–22, 2010. 10.1007/s11276-010-0276-9.

[98] R. Baumann, S. Heimlicher, M. Strasser, and A. Weibel, "A survey on routing metrics, tik report 262," tech. rep., ETH Zürich, 2006.